First International WWW Conference, May 1994

Writing Documents For Paper And WWW

A strategy based on FrameMaker and WebMaker

Bertrand Rousseau *rousseau@ptsun00.cern.ch*

Mario Ruggier *ruggier@ptsun00.cern.ch*

Programming Techniques Group,
CERN, ECP Division,
CH-1211 Geneva 23, Switzerland

The interest in having high quality maintainable versions of a document in both paper and WWW formats is growing dramatically as WWW is fast becoming the most important way of browsing and delivering information on the Internet.

This paper discusses the strategy based on FrameMaker, a desktop publishing package from Frame Technology Corporation, and the configurable translator WebMaker, written by the authors, for the simultaneous production of paper and WWW versions of a document. Advantages of this scheme are the WYSIWIG FrameMaker interface, easily maintainable WWW documents, and the possibility to obtain a WWW version of already existent Frame documents by simply supplying configuration information for their conversion.

The introduction provides some background information and outlines the strategy. The second section is a comparison of the two formats, feature by feature, outlining the behavior of WebMaker for each feature. The third section focuses on WebMaker and discusses the general user requirements and the WWW document generation process. The WebMaker configuration language is briefly discussed. The fourth section presents a list of recommendations for writers as a guideline for the simultaneous preparation of both the paper and WWW documents. A few words on future directions is given at the end.

1 Introduction

The recent phenomenal expansion in the popularity of WWW, while offering the exciting possibilities of heretofore unknown experiences, also demands that many documents are to be available in both paper and WWW form.

Current HTML authoring tools offer inadequate functionality for the comfortable development of Web documents. A good HTML editor, if one were to become available, may facilitate the creation of Web documents but it is unlikely that it could match the type-setting capabilities of desktop publishing packages such as FrameMaker [1]. Consequently a good WWW editor would still not be sufficient for those authors conditioned by the necessity of having both high quality printed and WWW versions of the same document.

1.0.1 Conventional documents on the Web

The advent of the World-Wide Web offers an essentially new medium for the dissemination of information. Such a distributed hypertext multimedia environment is likely to provide the right conditions for the evolution of a new document genre. Just as the novel, the most natural expression of the printed book medium, could not be conceived before the invention of cheap printing, neither can such a genre grow without the coming of age of WWW.

However, the evolution of a native WWW document genre does not mean that there is no place for the more conventional documents on the Web. Similarly, transposing this observation to printed documents, there is still a function and purpose for the numerous types of printed documents that could be much more useful when coupled to the functionality of WWW, e.g. extensively indexed reference manuals.

1.1 A solution

We have attempted to meet the demands for having both printed and WWW versions of a document by developing a strategy based on FrameMaker, a desktop publishing package from Frame Technology Corporation, and the configurable translator WebMaker, written by the authors, to generate a WWW version of a paper document formatted with FrameMaker. This strategy was first conceived to satisfy our own documentation needs [2]. The demand for a more generic solution was growing alongside the growth of the Web itself. WebMaker is the result of a generalisation of our strategy.

With the combination of FrameMaker and WebMaker it is possible to derive paper and WWW versions of a document, without compromise to either, from a single master source which would consist of:

- A FrameMaker document (single file) or a FrameMaker book (multifile).
- A configurable rules file for the mapping of the content body into HyperText Markup Language (HTML), the native language of WWW.

As there are no restrictions made on the layout and structure of the FrameMaker document, full advantage may be taken of FrameMaker's page formatting features to obtain the desired printed version. Similarly, as the rules file for mapping the content body into HTML is highly configurable, a WWW version of the document taking good advantage of the different functionalities of the medium may automatically be derived.

1.1.1 Added advantages

Such a scheme based on FrameMaker and WebMaker also offers the added advantages of:

- The FrameMaker WYSIWYG authoring interface with built-in drawing tools, spelling checker, numerous import and export filters, etc.
- The easy maintenance of unwieldy WWW multi-file document webs.
- FrameMaker documents that were originally written with no WWW intentions may be translated simply by adding configuration information for the mapping into HTML.

2 A comparison of printed and WWW documents

Printed and World-Wide Web documents are intrinsically different and preparation of a document for production on both these media must take into account these differences. Some documents are likely to be more effective in one of the two formats, for example bulky reference manuals.

This chapter compares the two media feature by feature and discusses briefly the corresponding behavior of WebMaker. We state first the basic criteria for the functionality of either format.

2.1 The criteria for functional documents

2.1.1 Paper

For a given document content, the validity of a representation of this on paper depends *entirely* on the physical appearance of the printed document.

To obtain a desired physical appearance FrameMaker is employed.

2.1.2 WWW

The criteria for a functional WWW document are dramatically different. In fact the author has extremely limited control on the details of the presentation as this is defined by the browsing software. A functional WWW document depends on:

1. The usefulness of the hypertext links contained in the document.
2. The provision of indices giving the required access modes.
3. The block size in which information is delivered, i.e. the size of the HTML nodes that make up the Web document; large nodes make for uncomfortable browsing while nodes that are too small may be too fragmented for the effective flow of ideas.
4. Navigation ease.
5. The blending of non textual elements such as graphics with the text.

WebMaker is employed to produce such a functional Web document from a Frame document and supplementary configuration information.

2.2 A feature by feature discussion

We here identify the major properties of both media and how each of these is handled by WebMaker.

2.2.1 Document structure

Superficially, the structure of paper documents is significantly different from that of WWW documents. Printed documents are deemed to be linear while hypertext is non-linear. Despite this, few printed documents are read in a linear fashion. This necessitates the provision of non-linear navigation aids such as tables of contents and alphabetic keyword indices and suggests that such documents are well suited to benefit from a hypertext environment.

Some printed documents are in fact essentially linear and should be consumed in a linear fashion, for example reading a novel in a comfortable chair. However, many conventional printed documents such as reference manuals, newspapers, indexed structures such as dictionaries, encyclopedias and so on, could in fact be much more usable in a hypertext format.

WebMaker can read configuration information to decide how to break up the text into the different HTML nodes. Both linear and hierarchical navigation aids may be automatically supplied, for example, links to the linear next and previous pages, links to the topmost node, to the parent node, to tables of contents, indices and so on. Non-hierarchal hypertext links to the same Web document or to other resources may also be included.

2.2.2 Textual building blocks

The basic language building blocks conveying information, i.e. the words, must provide the common ground between the two forms of a document. This is not to say that there are no subtle differences in the wording between two ideal paper and WWW representations of a document. However, without the restriction that the textual body is the same in both formats a comparison of the two media is meaningless and the existence of converters between them would not be possible.

WebMaker translates text word for word. For the minor cases where some differences in the text are desirable as, for example, in the case of the wording of this cross reference to the References section below, a simple solution is provided. Such subtle differences may be controlled by conditional text for paper and conditional text for WWW.

2.2.3 Unlimited character set

The character set available for printing is wide; with most systems additional typefaces may be incorporated as needed. The character set available in HTML is limited to ISO Latin 1, requiring special software for other sets, for example Japanese.

This limitation will be considerably less relevant, if not disappear altogether, with the implementation of HTML+ [3]. Until such time as this, WebMaker includes, in the generated Web files, an image representation of characters in FrameMaker that are not in ISO Latin 1.

2.2.4 References

WWW documents may reference information sources including other documents and active data sources such as on-line news conferences, mail and databases. References are not limited to the scope of a single document or file system. This is made possible by the use of Uniform Resource Locators, or URLs, which describe a data object and contain sufficient information for their retrieval. This is the fundamental WWW principle.

In comparison, references in paper documents redirect the reader to other marked (numerically or otherwise) locations in a document. References in WWW are more convenient as they do not require the reader to perform any action to retrieve referenced data other than a click of the mouse.

WebMaker translates all cross references within the same FrameMaker document into hypertext links. HTML hypertext links may be inserted anywhere in the text by a simple system that is completely independent of the hypertext facilities in Frame: A special FrameMaker marker (Type 25) containing the URL of the destination is inserted just before the anchor text, which is then assigned a dummy character format `HTMLLink`.

Cross references between FrameMaker documents. are not currently being translated (see Further work on page 9).

2.2.5 Graphics

Graphics in FrameMaker may be either native, imported by copying into file or imported by reference. Some WWW browsers can display images, normally in GIF, in a similar way as they display character entities within text.

WebMaker adopts the method implemented by Jon Stephenson von Tetzchner in the fm2html converter to translate graphics. All three FrameMaker graphics cases are translated by first extracting each one into a separate MIF file, then translating to PostScript, cropping and translating again to GIF (default format) or to another format specified by the user, per graphic, in the Frame master file. An image may also be specified to be the destination of hypertext links, in which case a clickable iconised image that points to the external image is inserted into the text. Other configurable actions regarding graphics are the format of external image files and the sizes, in terms of the original, of the generated icon and of the actual image.

2.2.6 Some elements exclusive to WWW

WebMaker also allows the inclusion of non textual WWW objects, such as graphics and animation that contain hypertext links, sound and interactive forms, in the FrameMaker master. Any text in the FrameMaker master tagged with the `HTML` conditional text tag is translated as raw HTML. Printing of this is suppressed if text conditional under `HTML` is hidden.

2.2.7 Mathematics, tables and figures

As HTML does not support mathematics, tables and figures, these may not be as such translated. WebMaker adopts a solution similar to the one first introduced by Nikos Drakos in the LaTeX2`HTML` converter [4]. The solution is to treat highly formatted objects for which there is no equivalent in HTML as pictures. Each one is therefore isolated into a separate MIF file, printed to a PostScript file, cropped and translated to GIF, and then put back in the generated Web document.

The implementation of HTML+ will allow a real translation of these objects.

2.2.8 Searchable index

WWW documents may offer powerful indexing features which is especially useful for reference material that is normally consulted for very particular information. The `ISINDEX` tag enables users to send a string of characters to a server for interpretation. The actions taken by the server depend on the selection of an external program running on the server. The implementation of forms in HTML+ significantly enhance this feature.

An author may specify whether a generated Web document is to be a searchable index.

2.2.9 The confines of a document

Paper documents are defined by physical size. HTML does not provide the means to mark the perimeter of a document or whether a *node*, or single HTML file, is part of a Web document or not.

WebMaker uses such paper intrinsic information to organise the generated Webs by inserting a navigation panel in each node. This helps a reader to know exactly which page of the document and which document are being browsed.

2.2.10 Page layout

Page layout information belongs to the realm of the printed paper. This may include information for headers, footers, page numbering, text columns and so on. This information is ignored by WebMaker.

2.2.11 Transient and accumulative contents

WWW documents may continually change without causing the need for a new *copy*, i.e. the document would still be accessed by the same URL. This implies that a WWW document may contain general information about a given subject that is constant as well as continuously changing information that can *always* be kept up to date.

Printed documents are frozen; modifications will necessitate a new physical copy.

This property of WWW does not translate to any WebMaker requirements. However, the maintenance of a regularly updated document is considerably simplified if the master is in FrameMaker and WebMaker.

3 WebMaker

WebMaker is a configurable translator that maps a Frame document onto an HTML Web. Its aim is to allow authors to take full advantage of both media.

3.1 General user requirements

We have identified above the more important differences and similarities between paper and WWW versions of a document and the criteria governing their functionality. We now express this information as the general requirements that the WWW document generator WebMaker should have.

1. WebMaker is able to convert any FrameMaker document file or files that make up a FrameMaker book.

   There may not be any restrictions made on the structure, layout or final presentation of the paper version of a document simply because a WWW version is to be produced. However, there is more than one way to achieve a desired structure, layout and final presentation of a document for paper. There is normally a ‘proper’ way and this is usually the way that FrameMaker was meant to be used. Please see the section Recommendations for writers on page 8.

2. WebMaker can map all possible FrameMaker entities to HTML entities.

   FrameMaker entities that WebMaker can currently translate are text made up from ISO Latin 1 characters, non ISO Latin 1 characters, anchored frames, mathematical formulae, tables, footnotes, cross-references, lists of contents, glossaries, indices, native graphics and imported graphics. WebMaker offers a strategy to map all of these entities into HTML.

3. WebMaker can be configured to resolve all issues relevant only to the WWW version of a document.

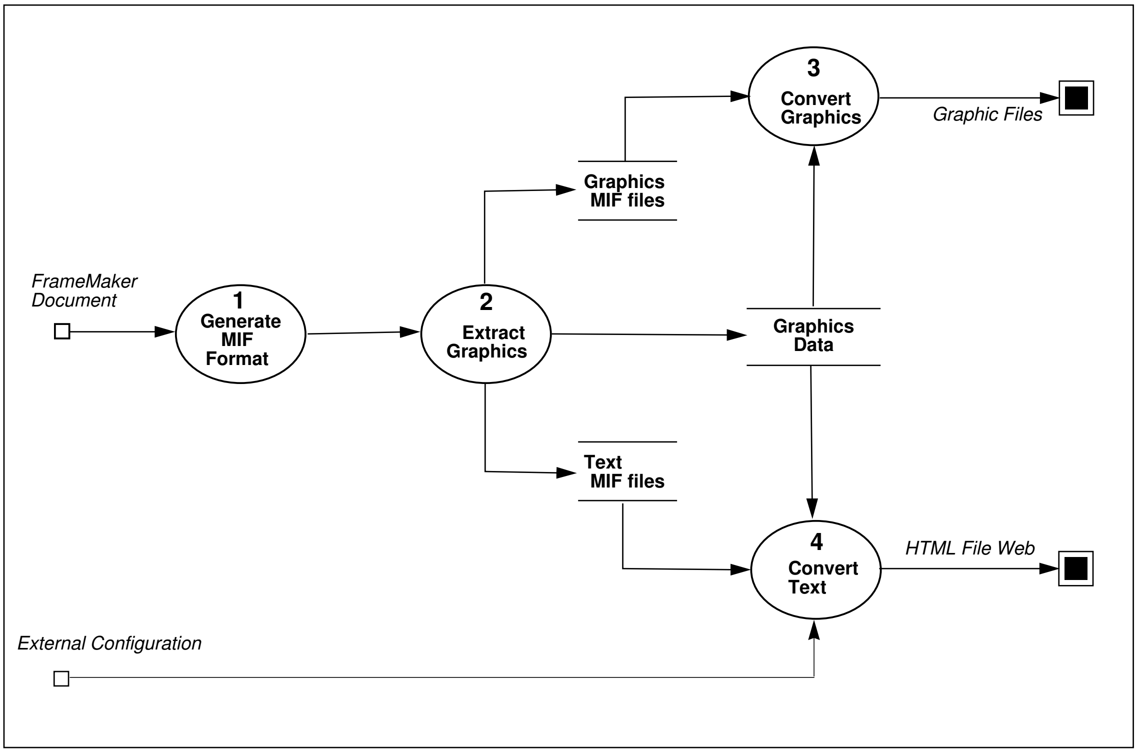
   Information that is foreign to the environment of the paper version of a document may be supplied by the user to WebMaker. This information is then used to generate the WWW document with the desired presentation and functionality. Typical such information may be how should the blocks of text be split up into the different HTML files, navigation options, search options, whether a graphic is to be inlined or not, and conditional text enabling the inclusion of special HTML features and/or the exclusion of contents intended only for printing.

The document *WebMaker User Requirements* [5] contains a complete list and explanation of the WebMaker user requirements.

3.2 The conversion process

The Data Flow Diagram in Figure 1 outlines the conversion process of WebMaker.

Figure 1



1. The binary files of a FrameMaker document are converted to Maker Interchange Format files.
2. Each such MIF file is then split into several MIF files and a data file:
    a. One MIF contains all the text of the original MIF.
    b. Each graphic, table and mathematical equation is put into a separate MIF.
    c. A data file retains necessary information about the relation between the text and each graphic, to be used for the re-integration of the images into the generated Web.
3. The graphic MIF files are converted to graphics files in the specified formats:
    a. Images are translated to GIF if they are to be inlined into the text.
    b. Images to be viewed externally are translated to a user selected format. Also, an author may specify the inclusion of the iconised image as an anchor pointing to the external image itself.
4. The MIF containing the text is converted to HTML according to an external configuration file and utilising information in the generated data file to re-integrate the graphical images.

3.3 The WebMaker configuration language

The rules for the mapping of a FrameMaker template into HTML are specified in the WebMaker configuration language. FrameMaker documents using the same template may therefore use the same mapping and more than one mapping may be developed for the same template, if this is desirable.

The configuration program is divided into three parts:

1. The rules governing WWW node generation.

   The aspects of a generated HTML node are described here. These include specification of a navigation panel, information about the HTML elements TITLE, ISINDEX, ADDRESS, etc. Each general type of node requires one set of node rules.

2. Paragraph format mapping rules.

   Rules specifying how to map each FrameMaker paragraph tag to an HTML construct, if actions different than the default are desired.

3. Character highlights mapping rules.

   Rules that associate a FrameMaker character format tag with an HTML character highlight tag.

Qualities of the configuration language are that it is easy to read, easy to write and it provides as much flexibility as HTML allows. HTML elements are referred to logically and not by raw HTML. This means that a programmer does not need to remember details of HTML syntax and guarantees the output of correct HTML.

3.3.1 Example

Suppose we want to translate the FrameMaker paragraph format `SpecialNote`. An example of one in FrameMaker may be:

A special note paragraph is enclosed between two reference frames, each containing a horizontal line.

We would like this paragraph to be translated to:

```
<HR>
<P><EM>A special note paragraph is enclosed between two reference
frames, each containing a horizontal line.</EM>
<HR>
```

The code in the configuration file that does this is:

```
PARAGRAPH SpecialNote is Normal
{
 actions
    {
      write(*,hrule());
      write(*,paragraph(emphasise(@text)));
      write(*,hrule());
    }
}
```

Paragraphs are assigned a type of paragraph that they are to be mapped into, in this case `Normal`. The '*' in the `write` statements tells WebMaker to write into the current node file. The predefined function `hrule()` specifies the HTML markup for horizontal rule. The predefined variable `@text` contains the text of the paragraph being currently translated. The predefined function `emphasise()` maps a string, the value of `@text`, to the emphasised character highlight. The predefined function `paragraph()` maps the value of the string parameter, `emphasise(@text)`, to a normal paragraph in HTML.

3.4 The hardware and software platforms

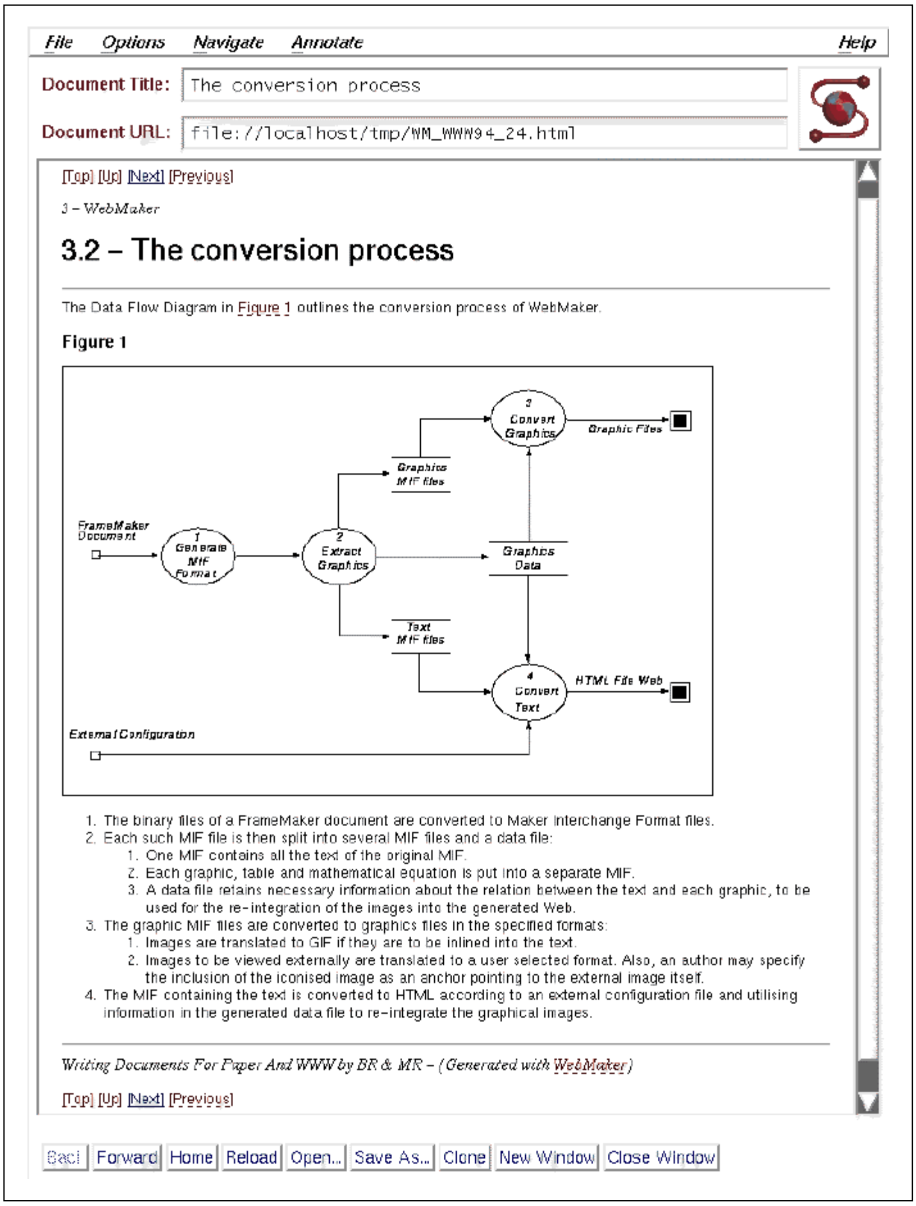
WebMaker is a set of lisp, awk and shell programs and runs on UNIX workstations. It is currently implemented to process MIF files generated by FrameMaker 3.0. The following free software is required:

1. xlisp 2.1e
2. gawk
3. ghostscript
4. pbmplus graphics package

3.5 A sample page

Figure 2 shows Mosaic's interpretation of the WebMaker generated WWW version of Section 3.2 of this paper.

Figure 2



4 Recommendations for writers

The rule of thumb for authors is that they should always attempt to organise a document in such a way as to maximise ease of maintenance. Invariably, this also makes for more translatable documents. This implies the abundant use of *Cross References*, formats from the *Paragraph* and *Character Format Catalogues*, and *Anchored Frames* i.e. frames anchored to a point in the text thus moving along with the text.

4.0.1 Use the format Catalogues

Although FrameMaker allows writers to highlight character and format paragraphs with locally defined properties, this is certainly not advisable; the maintenance of such documents is unnecessarily difficult. Furthermore, such character and paragraph formats cannot be translated logically according to a user specified configuration.

There is no limit on the numbers of formats that may be put into the FrameMaker paragraph and character catalogues, or in the number that WebMaker can handle. Writers are therefore encouraged to define and register as many paragraph formats as necessary. Global modifications may cause the loss of any existent local formatting information. Also, a writer can specify appropriately different actions to be taken for different format tags only if these are catalogued under different names.

4.0.2 Put Graphics into Anchored Frames

Frames that are not anchored to a point in the text are positioned absolutely on the page. This makes it impossible to position the graphical contents of the frame correctly in the generated Web. Because of this WebMaker ignores all objects not belonging to a Text Column. The only way to record the relative order of objects in a FrameMaker document is to put non textual objects in anchored frames.

4.0.3 Design the paper document generically

A given physical appearance for a printed document may be obtained in a multitude of ways, even when using the same software. For example in FrameMaker one may enter a reference to the Introduction by either typing the character string ‘see Introduction’ or by entering a real cross reference, ‘see Introduction’, that may then be updated automatically and contains information that enables a program to easily translate this to a hypertext link. However it would not be possible to discern any difference between the two references on the printed page.

Generically designed documents are easier to maintain and give better translation results.

5 Further work

The intention of the authors is to keep developing WebMaker by upgrading to FrameMaker 4.0, adding new features, refining existent features and extending to map onto HTML+ as it becomes standardized.

5.0.1 Existent problems independent of HTML

1. Sequential processing of paragraphs.

   WebMaker minimises the amount of document contents in memory at any point in time. This is done to guarantee that large documents may be reliably translated even on machines with few resources. The drawback is that WebMaker cannot look too far ahead to try and determine whether to close or open environments. Because of this some complex HTML constructs cannot be mapped onto. A refinement of this scheme is under way.

2. Text in graphics.

   Due to difference in resolution between the printed page and the computer screen, text in graphics is often unreadable. As a temporary solution WebMaker offers the option to scale up the size of translated graphics.

3. Graphics imported by reference in FrameMaker documents.

   Currently WebMaker is translating these in the same way as other graphics. However the author should have the option to not change the original graphic file and just add a link pointing to it from the appropriate place in the generated Web.

5.0.2 Missing features to be added

1. Cross references between FrameMaker documents.

   Cross references to points in other FrameMaker documents are currently not translated. The difficulty is determining the node and the name of the anchor to point the hypertext link that such a reference is mapped to. A solution has been devised but it remains to be implemented.

2. FrameMaker hypertext markers.

   These are currently not being translated. A scheme to select the ones that are useful and throw out the ones that are not is being designed.

3. Unanchored frames.

   In the section Recommendations for writers, authors have been advised to position all non-textual objects in the text by putting these in anchored frames. While this remains true, unanchored frames are still widely used. A solution is being devised to relate unanchored frames to a point in the text such that these may be translated into HTML.

5.0.3 Extending to HTML+

1. Mathematics, figures and tables.

   These will be translated properly as soon as an HTML+ browser becomes available. The module that parses mathematics in the MIF already exists. All that is needed is to reconfigure this to output mathematics marked up with HTML+, when this is finalised. The management of captions for figures and tables will also be solved and the contents of each cell of a table will be translated for what it is, i.e. text for text, mathematics for mathematics, graphics for graphics, ...

2. Character sets.

   WebMaker will incorporate the character set extension options of HTML+.

3. Other extensions...

   Other extensions, for example more HTML formatting options, will be incorporated and the configuration possibilities extended to allow as much flexibility to the author as possible.

6 References

[1] FrameMaker Reference Manual, Frame Technology Incorporated.

[2] The ADAMO Entity Relationship Programming System
http://www1.cern.ch/Adamo/ADAMO_ENTRY.html

[3] HTML+ (Hypertext markup format)
http://info.cern.ch/hypertext/WWW/MarkUp/HTMLPlus/htmlplus_1.html

[4] The LaTeX2HTML Translator
http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html/latex2html.html

[5] WebMaker User Requirements
http://www1.cern.ch/WebMaker/UserReq/Document.html