

Retaining hyperlinks in printed hypermedia document

Ernest Wan ^{*,1}, Philip Robertson ¹, John Brook ¹, Stephen Bruce ¹, Kristine Armitage ¹

Canon Information Systems Research Australia, 1 Thomas Holt Drive, North Ryde, N.S.W. 2113, Australia

Abstract

In this paper, we describe a method that allows a hypermedia document to retain its hyperlinks in the printed copy. The method associates the hyperlinks with cut-out tabs on the edges of the printed pages. A method for modelling the cut-out tabs and optimizing their assignment to the hyperlinks is discussed. We also describe a prototype authoring system that implements the method. © 1999 Published by Elsevier Science B.V. All rights reserved.

Keywords: Hyperlink; Hypermedia document; Cut-out tab; Printing; Cutting

1. Introduction

Hypermedia documents are computer-based documents that contain text, graphics, audio and video on pages that are connected by navigational links. The navigational links, often referred to as the hyperlinks, permit non-sequential or non-linear traversal of the document by the readers. A well-known source of hypermedia documents is the so-called World Wide Web (WWW) or simply the Web.

Hypermedia documents allow multiple views and efficient, non-linear exploration of information that are not possible with traditional books. On the other hand, the absence of an obvious linear structure and physical medium make it very easy to get lost in the hyperspace.

Hypermedia documents may be printed. Nevertheless, the hyperlinking functionality is typically lost in the printed copy. Most hypermedia documents especially those on the Web are intended for viewing

on the screen and designed to exploit the hyperlinking functionality. As a result, readability also suffers with the loss of the hyperlinks. For instance, removing the hyperlink to the definition of an unfamiliar term may make a description unclear to the readers.

Recently, the World Wide Web Consortium (W3C) has included elements in its Cascading Style Sheet Level 2 (CSS2) specification [2] to allow a Web page to have different styles for different output media. For example, different fonts, margins and colours may be specified for screen displaying and printing. Alternatively, a separate document especially designed for printing may be maintained. In this case, readability of the printed copy can be improved by re-organizing those parts of the documents that are affected by the loss of hyperlinks. Currently, no technique is available for preserving hyperlinks and, hence, the ability of non-linear browsing in printed copy.

Waters et al. have commented that “the things that hypertext can do best, such as linking... and quickly jumping back and forth among complex anchors

* Corresponding author.

¹ E-mail: {ernest,phil,johnb,steveb,kristine}@research.canon.com.au

and bookmarks, are unavailable in printed media". They considered this a major issue in Web printing, sometimes leaving the authors of Web pages with an unacceptable compromise [5]. Considering the inability to print hyperlinks a forgone conclusion, they suggested that "there is a fundamental dichotomy of usage between hypertext and printed information". Their view is commonly shared by other researchers and users of hypermedia documents which explains the lack of references on the subject. Clearly a method for retaining hyperlinks in printed copy would be significant for the printing of hypermedia documents.

In this paper, we describe a method that allows a hypermedia document to retain at least some of its hyperlinks in the printed copy. The method associates the hyperlinks with cut-out tabs on the edges of the physical pages. Techniques for presenting such association visually are discussed. It is obvious that the number of cut-out tabs and thus hyperlinks is limited by the physical dimensions of the pages. Methods for preserving edge space are described. We also present a method for modelling the cut-out tabs and optimizing their assignment to the hyperlinks. An experimental system that implements the approach and works on Web pages, is described. Finally, the potential applications of the technique and some practical issues are discussed.

2. Cut-out tabs as physical hyperlinks

Fig. 1 shows two typical examples of cut-out tabs and the terminology we are going to use for describing their various characteristics.

2.1. Traditional use of cut-out tabs

Cut-out tabs are sometimes used in traditional books and documents such as dictionaries and operation manuals to facilitate access to the various sections of the document. Cut-out tabs have also been used in some children books to allow non-sequential traversal of the material. The first application provides essentially an indexing function to the content. The later application although more often done for fun than other more subtle practical purpose is nevertheless hyperlinking. Typically, only a small number

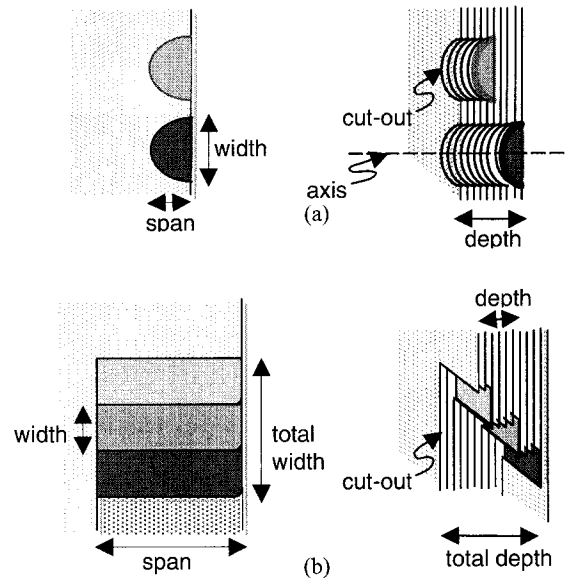


Fig. 1. Examples of cut-out tabs.

of cut-out tabs are used and seldom are they vital in delivering the content.

2.2. Hyperlink support

Cut-out tabs in paper documents and hyperlinks in hypermedia documents are analogous in both their functionality and appearance. When operating a cut-out tab, the reader places a finger or thumb within the cut-out tab, locates the first uncut page surface below the cut-out tab and opens that page. The action is not unlike clicking a hot-spot or anchor — the start point of a hyperlink — in a hypermedia document and traversing to the target page or the end point of the hyperlink (Fig. 2). In addition, like hot-spots, cut-out tabs are visually distinct from their surround.

A cut-out allows traversing the document in both the forward and backward direction. A cut-out tab that implements a forward (backward) link from an earlier page to a later page also provides a implicit backward (forward) path from the later page to the earlier page for free. The implicit return path can be used for hyperlinks linking information in the opposite direction. For instance, the two hyperlinks in Fig. 2 can share the same cut-out as shown in Fig. 3.

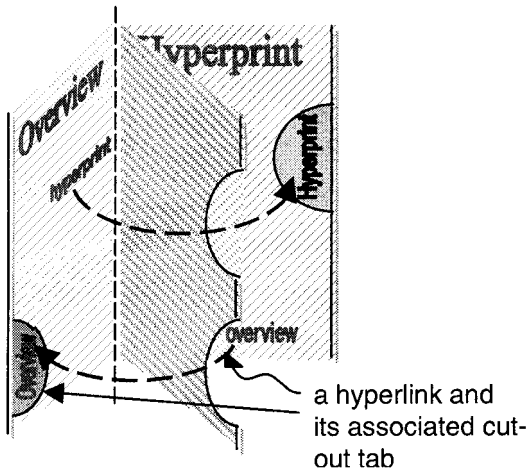


Fig. 2. Traversing hyperlinks in printed hypermedia document.

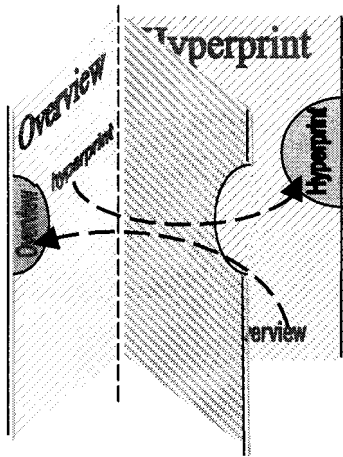


Fig. 3. Utilizing both the forward and backward paths of a cut-out tab for hyperlinks in opposite directions.

2.3. Styles and properties

The cut-out shows part of the linked page. The uncut surface reviewed is usually colour-coded or marked with a distinctive pattern or texture. Text label and/or icon can be placed within or alongside the cut-out tab to associate the cut-out tab with a hyperlink (Fig. 4).

2.3.1. Labelling

A cut-out tab is visible to and has an effect on all the pages the cut-out passes through. Placing the label/icon within a cut-out tab has the implication

that the hyperlink is relevant to all the intermediate pages. As an example, this type of labelling is suitable for implementing the “home” hyperlink found in many Web pages.

If the label or icon is placed alongside the cut-out tab, the label/icon can be removed or greyed out for those intermediate pages where the hyperlink is irrelevant (Fig. 5). By placing the label/icon alongside the cut-out tab, it also allows the cut-out tab to be shared by multiple hyperlinks that end at the same target page. Multiple labels/icons can be placed alongside the cut-out tab on a single page (Fig. 6). In addition, different labels/icons can be used on different pages.

Instead of placing labels/icons alongside the cut-out tab, text or graphical objects within the content can be used directly. In this case, a connector is used to connect the text/graphical object to the cut-out tab (Fig. 7). A colouring scheme or indexing scheme, although sometimes hard to decode when used alone, can also be used to associate the hot-spots to the cut-out tabs (Fig. 8).

2.3.2. Nested cut-out tabs

Related hyperlinks can be grouped to form a set of nested cut-out tabs of progressive size along a single axis (Fig. 9) freeing up edge space for other hyperlinks. When using a set of nested cut-out tabs, the user selects which hyperlink to traverse by simply moving the thumb or finger slightly to grasp the desired size of the cut-out. The set of rectangular tabs shown in Fig. 1(b) can be thought of as a special type of nested cut-out tabs.

It should be stressed that indiscriminate use of nested cut-out tabs may introduce artificial grouping of otherwise unrelated content and confuse the reader. Hence, nested cut-outs should only be used where a natural or logical relationship exists among the linked materials.

2.4. Conceptual grouping and perceptual layout

Nesting of cut-out tabs allows related linked materials to be conceptually grouped. Conceptual grouping of a hyperlink and its return path can also be achieved by merging a forward cut-out tab with its corresponding backward (or return) cut-out tab as in Figs. 3 and 10. A similar return path can be imple-

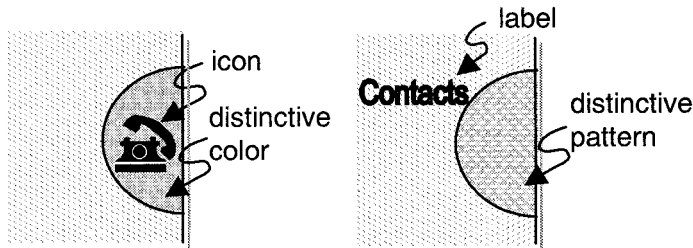


Fig. 4. Styles commonly used for cut-out tabs.

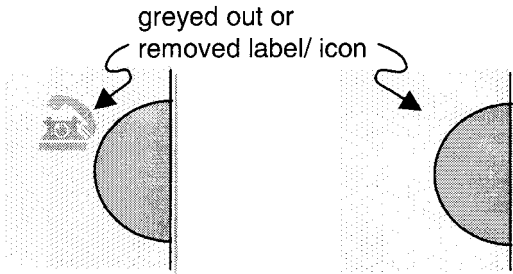


Fig. 5. "Disabled" cut-out tabs on pages where the associated hyperlinks are irrelevant.

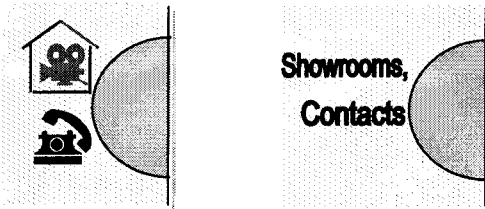


Fig. 6. Multiple hyperlinks that end at the same target page can share a cut-out tab by placing multiple labels/icons alongside the tab.

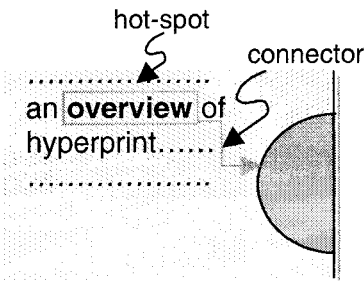


Fig. 7. Using connector to associate a hot-spot with a cut-out tab.

mented for the set of rectangular tabs in Fig. 1(b). The three unbounded edges of the printed document provide three natural dimensions for laying out hyperlinks of distinct functions or concepts. The

orthogonal axes of two adjacent edges can readily be mapped to two distinct ways of accessing the content. For instance, in a restaurants guide, one axis may be used for accessing restaurants by location and the other axis by cuisine. In the project plan of Fig. 11, the left/right edge is used for accessing the descriptions of the various products and the bottom edge is used for accessing the development plans.

2.5. *Printing and cutting*

2.5.1. *Duplex printing*

For a language such as English that reads from left to right, the right edge of the printed hypermedia document will be used for forward links and the left edge for backward links. Backward linking to a right-hand page has to use a cut-out tab that ends at the opposite left-hand page. If the document is printed double-side, then forward linking to a left-hand page also has to be done through a cut-out tab that ends at the opposite right-hand page (Fig. 12).

There are several implications. First of all, to support backward links, duplex printing is required (for printing the associated cut-out tabs) even if the content is printed only on one side. Secondly, when referring to the "start/end page" of a tab, we actually imply the start/end page and its opposite page. If the numbering of the pages is known before the optimization process, the conditions for merging the tabs should be relaxed accordingly. Thirdly, if the content is printed on both sides, a clearer means of associating cut-out tabs with hot-spots is especially important as a cut-out tab may become associated with content on the opposite page.

2.5.2. *Paper alignment and colour reproduction*

As a cut-out tab may pass through a number of pages, very precise paper alignment is required of

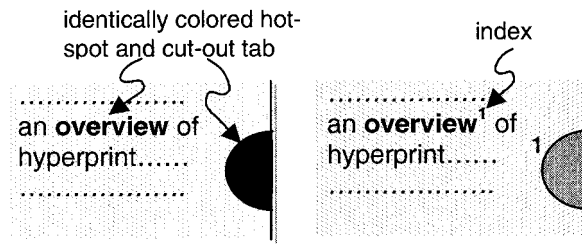


Fig. 8. Using colour and index to associate hot-spots with cut-out tabs.

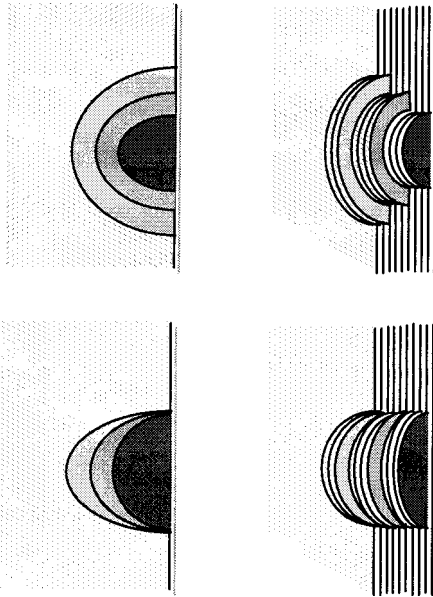


Fig. 9. Nested cut-out tabs.

the printer/cutter. If the hot-spot and cut-out tabs are colour coded, the colour printer used must be capable of reproducing the colour faithfully so that the set of colours carefully selected on the screen do not become indistinguishable on paper.

3. Cut-out tab model

A physical hyperlink from a hot-spot on page **S** to a target on page **T** can be represented by a data structure comprising of the one-dimensional vector \vec{ST} and the location of the target page, that is, the page number of **T**. Since the cut-out tabs that implement the hyperlink can only be in either a forward or a backward direction, only bi-dimensional vectors

are involved. The same basic data structure is used to represent the cut-out tabs each of which may be associated with multiple hyperlinks. Hereafter, depending on the context, the notation **ST** is used to denote a hyperlink or a cut-out tab from page **S** to page **T**.

Additional attributes such as the location of the tabs on an edge, the shape of the tabs, the colours and labels used by the tabs, etc., are required when rendering the tabs. Default values or author's inputs can be used for the value of those attributes that are not set by the optimization process. The author should be allowed to examine the results of the optimization process and make adjustments as required.

As mentioned earlier, a cut-out tab running from **S** to **T** has to pass through all the pages in between and is visible — although not necessary required by — those pages. Appropriate presentational style will be used to distinguish an inactive tab from an active one, for instance by removing or greying out the label of a tab when it has no associated hyperlink. An array must be added to the data structure of the cut-out tab to store the status of the tab for the intermediate pages. In case the tab is shared by multiple hyperlinks, a second array is used to store the labels/icons for the intermediate pages.

4. The optimization process

4.1. Overviews

The optimization process starts with a set of cut-out tabs each representing one hyperlink to be implemented physically. The set of cut-out tabs is then gradually reduced by combining cut-out tabs which can serve multiple hyperlinks.

Other physical attributes such as the size of the cut-out tabs and the page as well as certain system

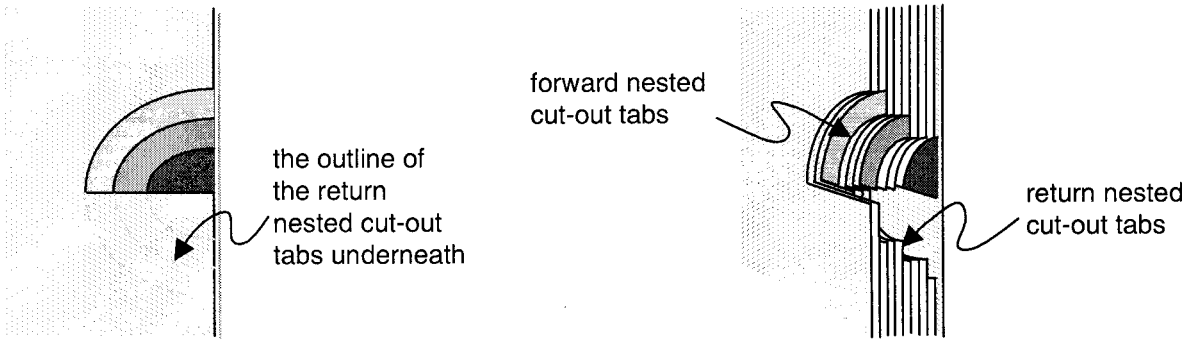


Fig. 10. Merging a forward nested cut-out tabs with its nested return tabs.

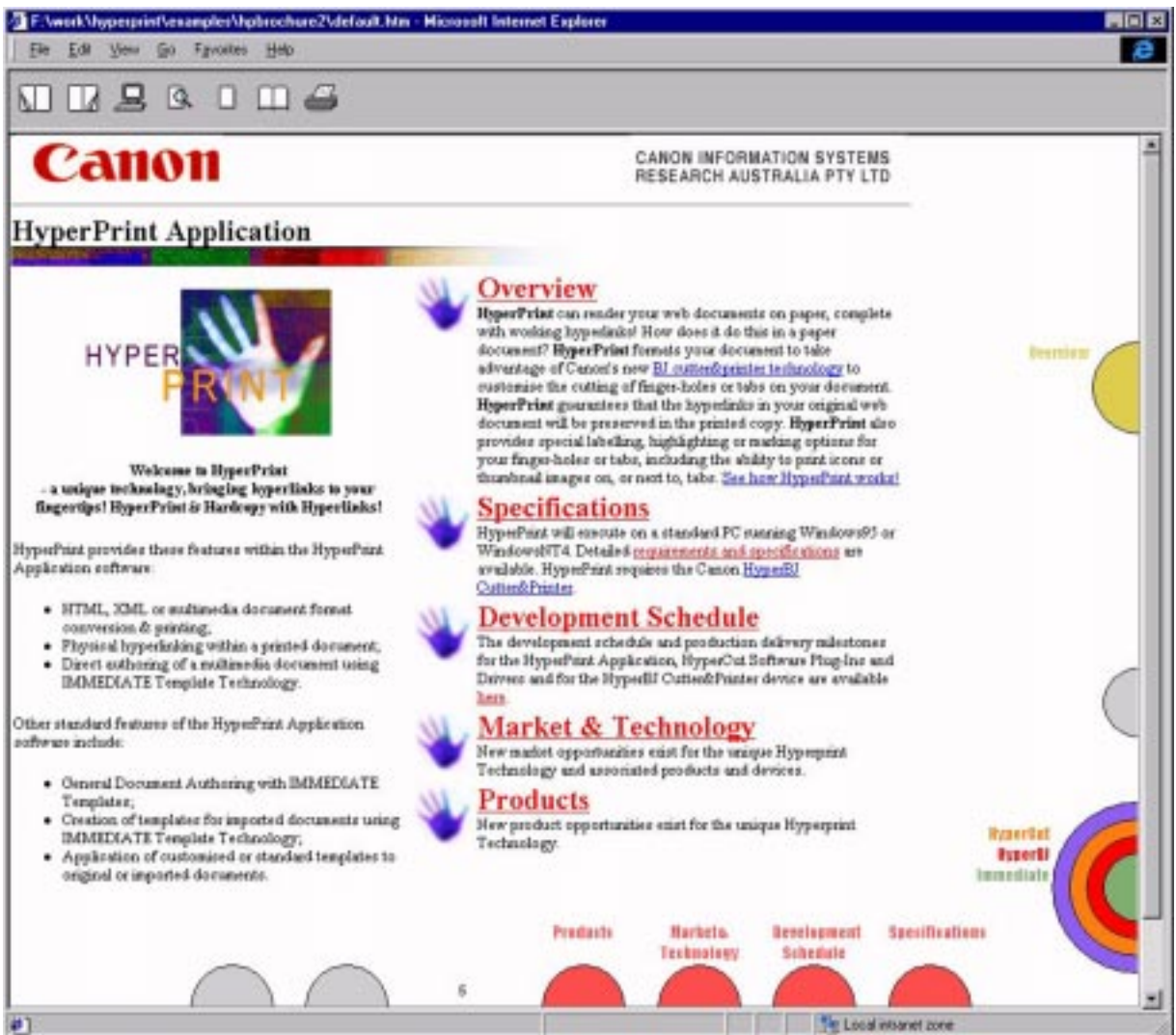


Fig. 11. The screen output of a project plan in print preview mode.

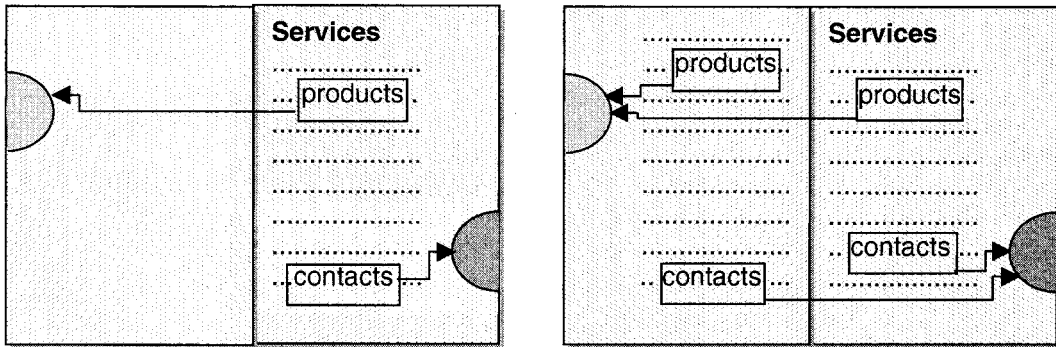


Fig. 12. Duplex printing is required to support backward links. Clear association of cut-out tabs and hot-spots is especially important when the content is printed on both sides.

parameters such as the edges to be used for tabs, the maximum number of tabs that can be nested along a single axis and the density of the tabs along an edge are required when “optimizing” (in a perceptual sense) the placement of the cut-out tabs.

4.2. Graphical notation

The graphical notations of Fig. 13 will be used to depict the cut-out tabs diagrammatically. The graphical notations can also be used in the graphical user interface (GUI) of an authoring system or editing tool to visualize the assignment and the physical arrangement of the tabs.

4.3. Optimizing the number of tabs

The optimization process involves the following basic steps.

- (1) Define the set of hyperlinks that will be implemented with cut-out tabs.
- (2) Assign a cut-out tab to each hyperlink.
- (3) Reduce the number of tabs by sharing the tabs among the hyperlinks.
- (4) Reduce the number of axes by nesting tabs.
- (5) Reduce the number of axes by aligning non-overlapping tabs.

The optimization steps can be automated. However, editing tools that allow the author to adjust the resulting hyperlinks/cut-out tabs assignment interactively are desirable.

In the next section, we will describe the optimization process in greater details using a restaurants guide example.

5. A restaurants guide example

Consider a restaurants guide that lists restaurants in a number of suburbs according to their specialty. The root (or starting) page of the restaurant guide is denoted as \mathbf{R} , followed by the restaurant directories for m types of cuisines \mathbf{D}_i , $i = 1, \dots, m$. Then come the suburb sections, one for each of the n listed suburbs. Each suburb section has an overview page \mathbf{S}_j and m cuisine sub-sections \mathbf{C}_{ij} , $i = 1, \dots, m$, one for each of the m cuisines where $1 \leq j \leq n$. The cuisine sub-sections contain the restaurants' details. In summary, the document has the structure:

$$\mathbf{RD}_1\mathbf{D}_2 \dots \mathbf{D}_m \mathbf{S}_1 \mathbf{C}_{11} \mathbf{C}_{21} \dots \mathbf{C}_{m1} \dots \mathbf{S}_n \mathbf{C}_{1n} \mathbf{C}_{2n} \dots \mathbf{C}_{mn}$$

5.1. Defining the hyperlinks

The hyperlinks defined are listed in Table 1. Each of the hyperlinks listed in Table 1 is assigned a cut-out tab. Additional hyperlinks that link each directory entry to the restaurant may be defined for the display version.

The set of cut-out tabs is initially given by:

$$\mathbf{H}_0 = \bigcup_{i=1}^m \{\mathbf{RD}_i\} \cup \bigcup_{r=1}^m \bigcup_{\substack{s=1 \\ s \neq 1}}^m \{\mathbf{D}_r \mathbf{D}_s\} \cup \bigcup_{i=1}^m \bigcup_{j=1}^n \{\mathbf{D}_i \mathbf{S}_j\} \\ \cup \bigcup_{j=1}^n \bigcup_{i=1}^m \{\mathbf{S}_j \mathbf{D}_i\} \cup \bigcup_{j=1}^n \{\mathbf{RS}_j\} \cup \bigcup_{j=1}^n \bigcup_{i=1}^m \{\mathbf{S}_j \mathbf{C}_{ij}\} \\ \cup \bigcup_{j=1}^n \bigcup_{r=1}^m \bigcup_{\substack{s=1 \\ s \neq r}}^m \{\mathbf{C}_{rj} \mathbf{C}_{sj}\} \cup \bigcup_{j=1}^n \bigcup_{i=1}^m \bigcup_{r=1}^m \{\mathbf{C}_{rj} \mathbf{D}_i\}$$


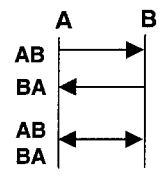
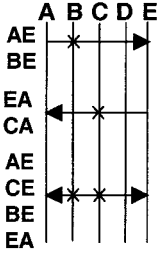
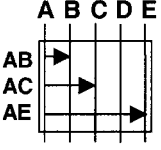
	<p>Page group: A vertical bar denotes a page group, that is, a set of one or more consecutive pages. The identifiers above the vertical bar denote the end-points of the hyperlinks that start or end at the page group, that is, the anchors and targets defined within the page group.</p>
	<p>Tab: A cut-out tab is denoted by an arrow. A left pointing arrow represents a forward link and a right pointing arrow represents a backward link. An arrow with arrow head at both ends represents a bi-directional tab that provides both a forward and a backward link. The arrow is labelled. The label indicates the hyperlinks that are associated with the tab. The label has the format <i>ST</i> where <i>S</i> is the identifier of the anchor and <i>T</i> is the identifier of the target.</p>
	<p>Shared tabs: To denote a cut-out tab that is shared by a set of hyperlinks, a cross is placed at the anchor page of each hyperlink except the outermost one along the arrow that represents the tab.</p>
	<p>Nested tabs: A set of arrows enclosed in a rectangular box denotes a set of nested cut-out tabs.</p>

Fig. 13. Graphical notations of cut-out tabs.

a total of $n(1 + 2m + 2m^2) + m^2$ tabs.

Repetitive structures and simple hyperlinks linking various sections of the restaurants guide are used in the example to make the optimization process easier to understand and its effectiveness easier to

demonstrate. The optimization method can be applied to documents with arbitrary cross-reference hyperlinks, although the physical constraints of the printed page might prevent some of the optimized cut-out tabs from being implemented.

Table 1
Hyperlinks in the Restaurants Guide example

Hyperlinks	Description
$RD_i, i = 1, \dots, m$	from the root page to each cuisine directory
$D_r D_s, r, s = 1, \dots, m, r \neq s$	from each cuisine directory to another cuisine directory
$D_i S_j, i = 1, \dots, m, j = 1, \dots, n$	from each cuisine directory to each suburb section
$S_j D_i, i = 1, \dots, m, j = 1, \dots, n$	from each suburb section to each cuisine directory
$RS_j, j = 1, \dots, n$	from the root page to each suburb section
$S_j C_{ij}, i = 1, \dots, m, j = 1, \dots, n$	from each suburb section to each of its local cuisine sections
$C_{rj} C_{sj}, r, s = 1, \dots, m, r \neq s, j = 1, \dots, n$	from each cuisine section of a suburb to the other cuisine sections of the same suburb
$C_{rj} D_i, r, i = 1, \dots, m, j = 1, \dots, n$	from each cuisine section of a suburb to the cuisine directories

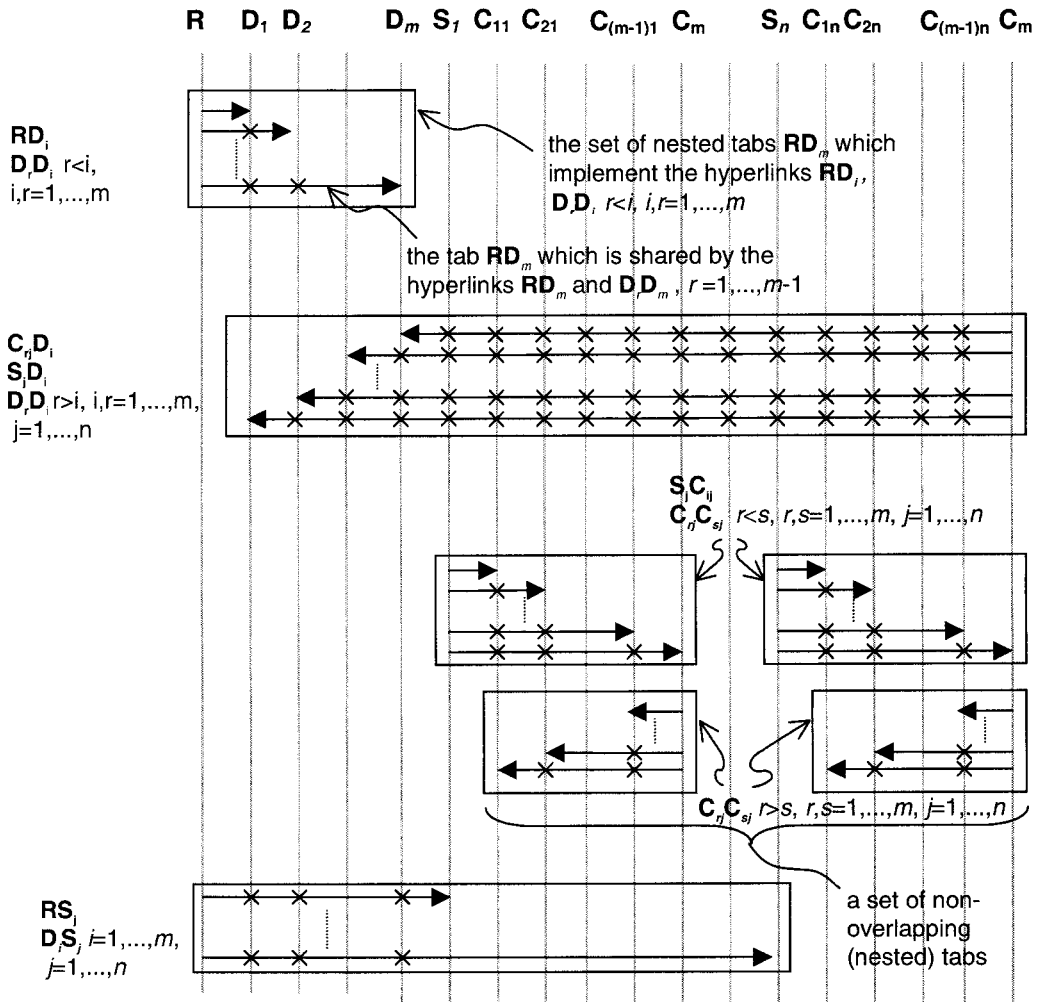


Fig. 14. The final tab assignment for the Restaurants Guide example.

5.2. Sharing cut-out tabs

A cut-out tab can be used for multiple hyperlinks that end at the same page as the tab. In fact, a tab **AB** can potentially be used for any hyperlink that starts at **A** or at a section between **A** and **B** and ends at **B** provided that the tab is properly labelled on all the pages it passes through and that such use is not confusing to the reader. For instance, referring to the final tab assignment in Fig. 14, the tab RD_m running from **R** to D_m is shared by hyperlinks RD_m and $D_r D_m$ where $r = 1, \dots, m - 1$. In addition, a tab **AB** leading from a page **A** to a page **B** provides an implicit return path from page **B** to page **A**. Hence,

the number of tabs can be reduced by merging tab pairs that have their two tabs start on the page the other tab ends.

By sharing tabs among hyperlinks, the original set of cut-out tabs is reduced to:

$$\begin{aligned}
 H_1 = & \bigcup_{i=1}^m \{RD_i\} \cup \bigcup_{s=1}^m \{C_{mn} D_s\} \cup \bigcup_{j=1}^n \{RS_j\} \\
 & \cup \bigcup_{j=1}^n \bigcup_{s=1}^m \{S_j C_{sj}\} \cup \bigcup_{j=1}^n \bigcup_{s=1}^{m-1} \{C_{mj} C_{sj}\}
 \end{aligned}$$

The total number of tabs is now $2nm + 2m$.

5.3. Nesting cut-out tabs

Related tabs are nested to reduce the total edge space required by the tabs. A set of nested tabs should link materials that have a natural or logical relationship. Such relationship may be inferred from the structure of the document and other information provided by the author. For instance, the set of nested tabs \mathbf{RD}_m in Fig. 14 are used for accessing the m restaurant directories each for a different cuisine. In this case, nesting the tabs reduces the number of axes required from m to 1.

The innermost unions of each of the terms of \mathbf{H}_1 can be nested. Assuming that the number of suburbs n and the number of cuisine m are small enough that nesting n and m cut-out tabs is visually acceptable, after nesting the tabs, the total number of (nested) tabs becomes $2n + 3$.

5.4. Aligning axes

The last two terms of \mathbf{H}_1 contain two sets of non-overlapping (nested) tabs

$$\bigcup_{s=1}^m \{\mathbf{S}_j \mathbf{C}_{sj}\}, \quad j = 1, \dots, n$$

and

$$\bigcup_{s=1}^{m-1} \{\mathbf{C}_{mj} \mathbf{C}_{sj}\}, \quad j = 1, \dots, n$$

Each of the two sets can share a single axis. As a result, the $2n + 3$ (nested) tabs require only five axes.

The final tab assignment is shown in Fig. 14. Fig. 15 shows the screenshots of a restaurants guide example. The Web document was created using Hyperprint — a prototype authoring system which we are going to describe in the next section. All Hyperprint documents are provided with a tool bar in the top frame to allow the user to flip through the document and switch between display and print preview modes.

6. Hyperprint

An authoring system called Hyperprint is being developed. The system creates Web documents that

can be printed and cut to support hyperlinks in their printed copies. It implements the optimization procedure we outlined in the previous section. The first prototype of the system allows the author to:

- (1) import a set of HTML (HyperText Markup Language) Web pages to a document — the original HTML pages do not need to be organized in pages of similar length;
- (2) divide an imported HTML page into multiple HTML pages each printed on one physical page — this is done automatically at the time the HTML page is imported;
- (3) specify the tree structure of the document;
- (4) select the hyperlinks to be physically implemented;
- (5) optimize the hyperlinks/cut-out tab assignment;
- (6) re-position the tabs and specify their styles;
- (7) save the final HTML pages.

Fig. 16 shows the user interface of Hyperprint and the main steps of composing a Web document with Hyperprint using a project plan as an example. Fig. 17 shows the printed (and cut) copy of the project plan produced. Unlike the restaurants guide example, the project plan does not have repetitive structures and has arbitrary cross-reference hyperlinks.

The generated HTML pages can be displayed by browsers such as the Internet Explorer 4 (IE4) that support CSS1 [1] and the Document Object Model (DOM) [3]. The HTML pages use applets for rendering the cut-out tabs. The cut-out tabs are functional both on screen and on paper. On screen, clicking the tab causes the target page to be displayed.

The parameters of the cut-out tabs are stored as Javascript variables in a “parameter” script. The “parameter” script is imported into each of the HTML pages together with a set of library scripts. Among other things, the library scripts provide a set of functions that the HTML pages may call to create HTML elements for the cut-out tabs.

IE4 can be used for printing the HTML pages of the document. At the moment, a graphic design package is used to drive a cutter and produce the cut-out tabs. The cut pages are then wire bound. A utility is being developed for printing and cutting the documents using a printer/cutter.

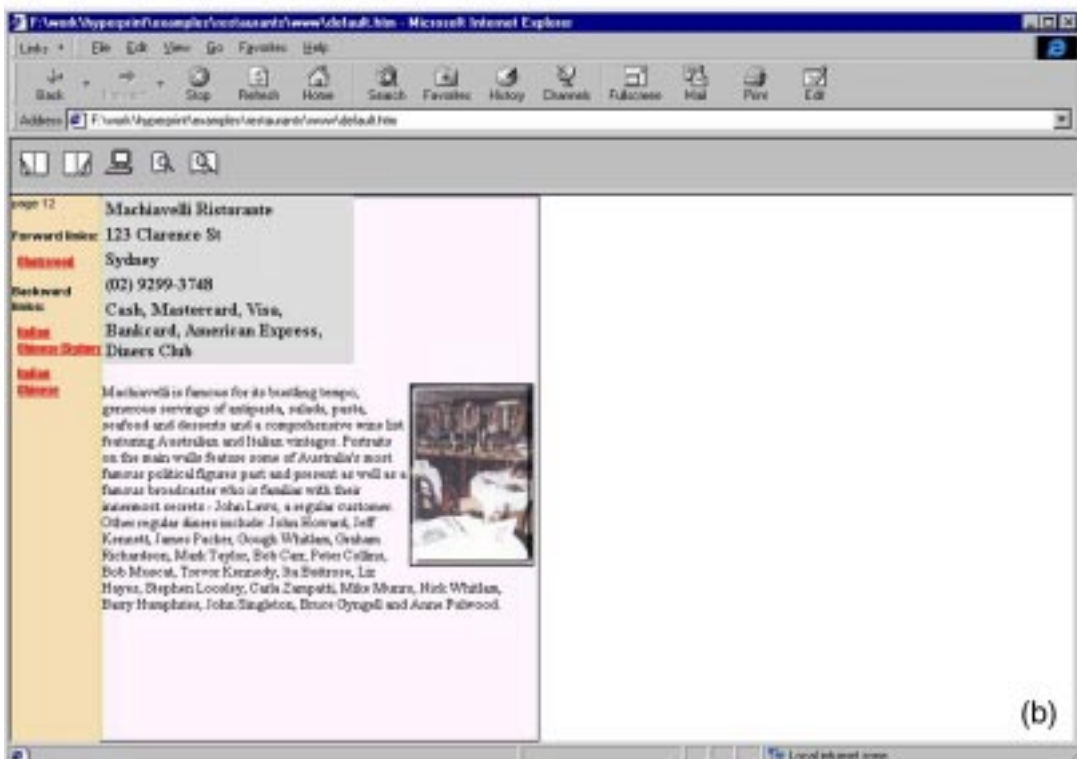
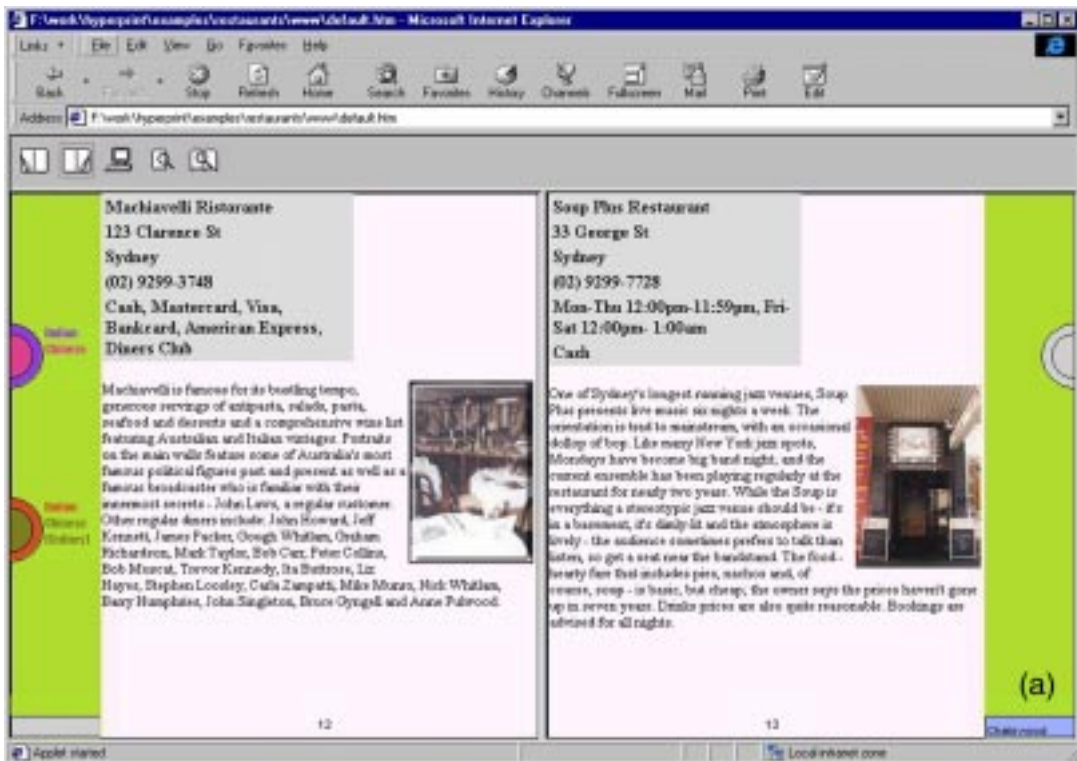


Fig. 15. Screenshots of the restaurants guide example, in print preview mode (a) and display mode (b).

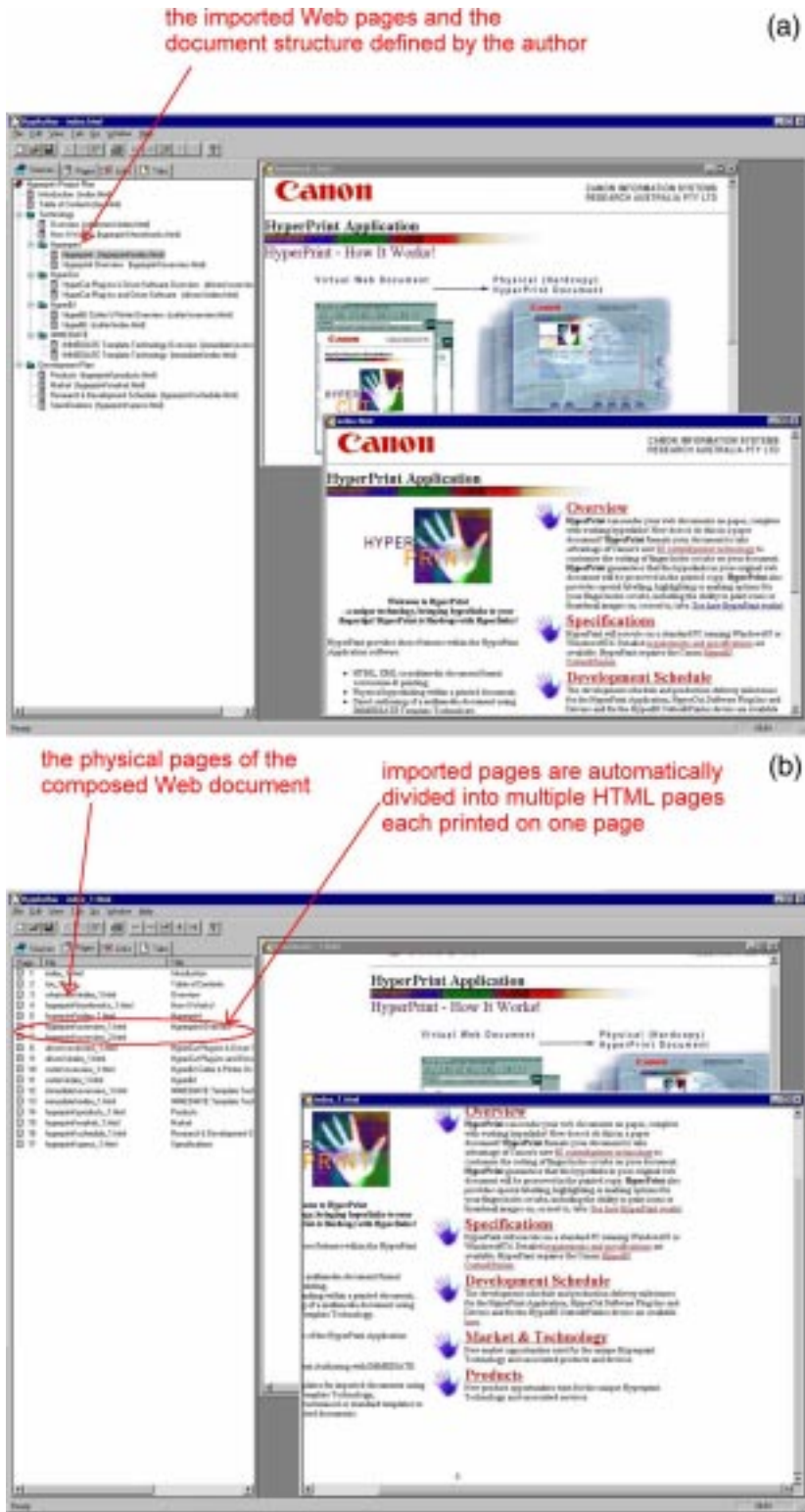


Fig. 16. The user interface of Hyperprint and the main steps in composing a Hyperprint document from a set of Web pages. (a) Importing the source pages and defining the document structure. (b) Dividing the source pages into multiple HTML files each printed on one physical page.

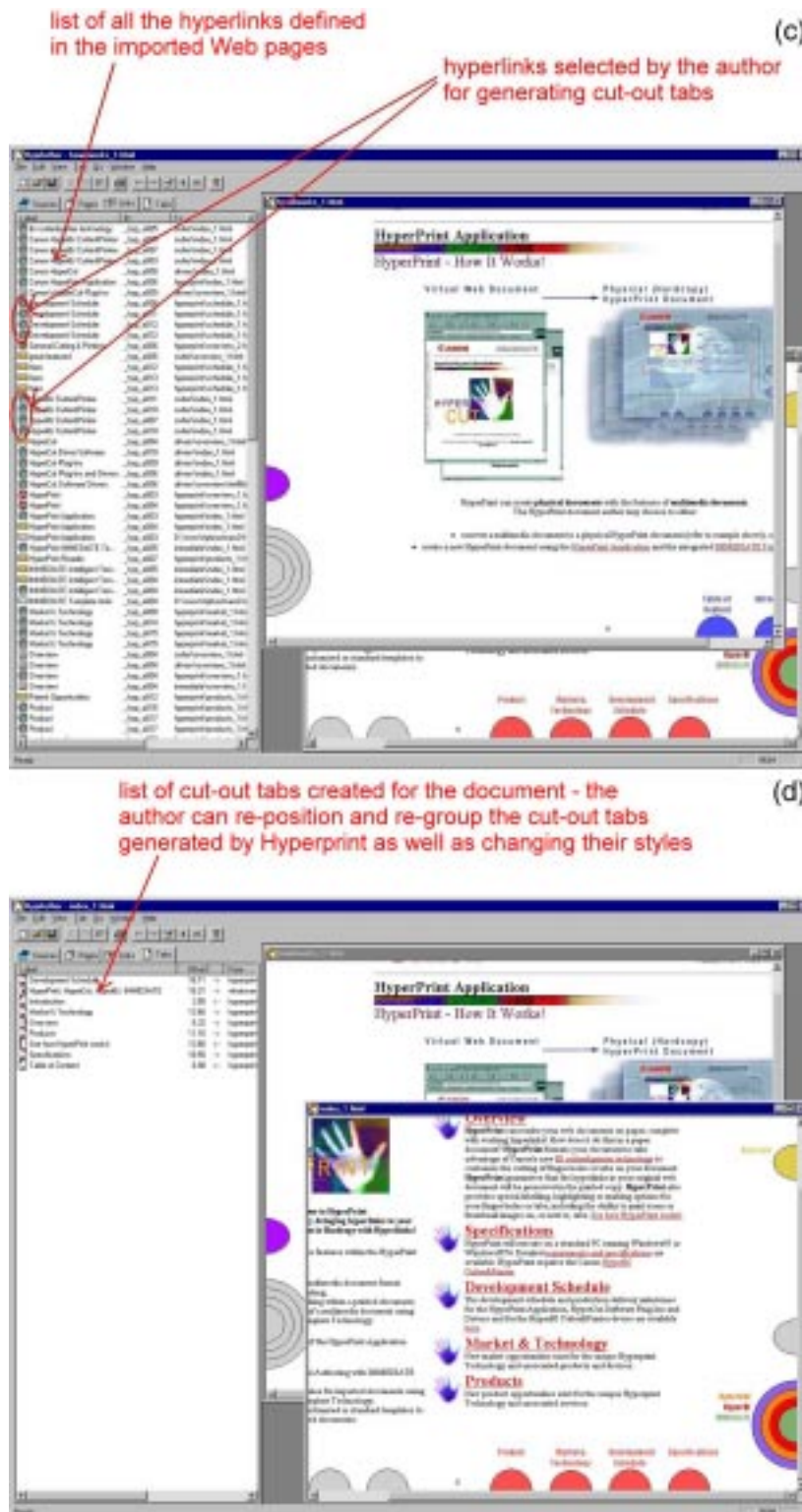


Fig. 16 (continued). (c) Selecting the hyperlinks to be implemented as cut-out tabs. (d) Generating the cut-out tabs and defining their styles.



Fig. 17. The printed (and cut) copy of a project plan created with Hyperprint.

7. Discussion

Our method allows Web documents and other hypermedia documents to be published in printed copies exploiting both the advantages of the print media and a hyperlinked form. We feel that it is most useful for documents that possess some of the following properties:

- the document is well-structured, e.g. a technical report;
- the document has a well-defined scope and function, e.g. an operation manual;
- the document contains related material that can be read in different orders, e.g. a reference manual;
- the document contains related material that provides multiple views, e.g. a maintenance manual that is used by technicians specialized in different areas;
- the document contains material that is suitable for content-based exploration;
- the document contains material that is mostly locally linked, that is, most of the cut-out tabs required are shallow cut-out tabs that can share their axes rather than deep cut-out tabs that cut through a large number of unrelated pages;
- the document contains material that requires frequent cross-reference to content on different pages, e.g. a financial report;
- the document contains personalized and frequently updated content, e.g. online subscriptions and program guides; the particular preference and interest of the individual helps to limit the number of hyperlinks required;

- hardcopy is an essential or the preferred recording/communication media, e.g. business plans.

At this stage, we have concentrated on developing tools for Web documents using specific examples. Our experience so far has revealed a number of practical issues.

- (1) The printer used must be capable of accurately aligned duplex printing so that the tabs can be cut precisely.
- (2) The printer used must be able to produce reasonably accurate colour as the cut-out tabs are usually colour coded.
- (3) Reasonably thick paper is required so that the pages are easier to flip and the cut edges are more durable. Nevertheless, we found that the thicker paper used for colour printing is often adequate for the purpose.
- (4) The printed copy has to be bound properly for the cut-out tabs to be usable. However, we found that even common folders that use locking rings produce quite good results.
- (5) A portion of the edge space should be reserved and not used for cut-out tabs so that the ease of turning pages in paper documents is preserved.

8. Conclusions

We have described a method that allows a hypermedia document to retain its hyperlinks in the printed copy. The method associates the hyperlinks with cut-out tabs on the edges of the printed pages. We discuss several presentational styles for the cut-out and their associated properties. A method for modelling the cut-out tabs and optimizing their assignment to the hyperlinks was also presented. We also briefly described a prototype authoring system, Hyperprint, that we have built. The system allows an author to construct Web documents that can be printed and cut to support hyperlinks in their printed copies. The more general problem of representing dynamic content such as audio, video and animation of multimedia document on paper media requires further research.

Defying the critics, paper documents have continued to survive and proliferate in the electronics age. Ted Nelson has described today's hypermedia as "tangled and limited", and that it has "no deep structure" and has "brought us interconnection but not understanding" [4]. In contrast, traditional paper documents usually have a well-defined structure and one expository sequence. Our method requires the author to organize his/her hypermedia document in the same way as a traditional paper document and not just interconnecting content. It also requires the author to put in some thought as to which hyperlinks are more important to the readers and, hence, should be implemented as cut-out tabs in the printed copy. However, the additional effort will lead to easier and better reading and should not be seen as a burden. Moreover, authoring tools such as Hyperprint can greatly reduce the effort involved.

We have not conducted usability trials for different applications. Areas that we feel have great potential include:

- marketing brochures of products/services,
- program guides,
- online subscriptions,
- information brochures such as facilities guide, restaurants guide, tourist guide, etc.,
- project/business plans and reports,
- manuals, and
- children's books.

Documents with a standard structure, such as project plans and business reports, allow template-based authoring which simplifies layout optimization.

References

- [1] Cascading Style Sheets Level 1 Specification, available at <http://www.w3.org/TR/REC-CSS1-961217.html>
- [2] Cascading Style Sheets Level 2 Specification, available at <http://www.w3.org/TR/REC-CSS2/>
- [3] Document Object Model Level 1 Specification, available at <http://www.w3.org/TR/PR-DOM-Level-1/>
- [4] T. Nelson, The Future of Information, limited distribution by ASCII Corporation, Tokyo, 1997, available at <http://www.sfc.keio.ac.jp/~ted/INFUTscans/INFUTscans.html>
- [5] S. Waters, S. Butler and R. MacMillan, Internet and Intranet Printing, available at <http://www.w3.org/Printing/iiprint.html>



Ernest Wan has a Ph.D. in Computer Science from the Australian National University. He is a Research Staff Member at Canon Information Systems Research Australia. His research interests include neural networks, knowledge-base and database systems, and internet-related technologies.



Philip K. Robertson is General Manager of Research and Development at Canon Information Systems Research Australia. He has a PhD in Computer Science from the Australian National University, and has published widely in image processing, graphics, colour and visualization. He is on the editorial board of IEEE TVCG, IEEE CG&A, and Computers and Graphics, and has served on program committees for

international conferences in these fields.



John Brook graduated from Canterbury University, New Zealand in 1983. He is currently working for Canon Information Systems Research Australia. His research interests include video and multimedia systems and applications, and internet-related activities.



Stephen Bruce received his B.D. in 1992 from the University of Technology Sydney in Visual Communication. Currently he is working as a Graphic Designer at Canon Information Systems Research Australia.



Kristine Armitage received her BA in Visual Arts degree from the University of Western Sydney, Nepean, N.S.W. in 1994 and at the same institution an Associate Diploma in Graphic Design and Photography in 1996. She has been part of the art department of Canon Information Systems Research Australia for three years designing promotional material and artwork for software developed by the company.