A STUDY INTO ELECTRONIC BOOK DESIGN AND PRODUCTION: HYPER-BOOK AND THE HYPER-BOOK BUILDER

Thesis submitted for the degree of Doctor of Philosophy

by

Nadia Catenazzi

University of Strathclyde

Strathclyde Business School

Department of Information Science

December 1993

Acknowledgements

Many thanks to Forbes Gibb, my supervisor, who followed my work with unusual patience.

Many thanks are also due to all the staff and colleagues who have provided both technical and human support: the group at the Joint Research Centre of CEC, JRC-Ispra, (Environmental Informatics Unit, Environment Institute), Italy; the Information Science Department at the University of Strathclyde, UK.

Many thanks to electronic mail, which was so useful for communications and contacts.

Special thanks to Lorenzo, for his help and support throughout this "adventure."

Last but not least, an acknowledgement is due to the Commission of European Communities (CEC), who provided financial support throughout most of my period of study.

Contents

Abstract

This work is concerned with research in the field of electronic publishing and, more particularly, of electronic books. It concentrates on an analysis of the stages of development of an electronic book from the acquisition of the source material to the actual use of electronic books, with particular attention to the interface design issues.

The main objective of this research is to demonstrate the appropriateness of adopting the paper book metaphor for representing electronic books, which are specifically designed to be parts of an electronic library. In addition, these electronic books are produced semi-automatically using a flexible publishing environment.

A model for an electronic book (*hyper-book*) and an environment which supplies tools for generating electronic books (*hyper-book builder*) have been defined and successively developed. *Hyper-books* are produced by importing electronic texts (in SGML or ASCII format) into an *empty template*, and by applying a number of tools in order to provide appropriate layout and reader services. *Hyper-books* incorporate hypertext features (e.g. history facilities and links), paper book features, and some other computer facilities (e.g. search functions). Design principles for *hyper-books* have been partially deduced from paper publishing and from experiments with electronic books.

An application in the environmental domain (in particular, in the area of Global Change) was developed and used in order to evaluate the *hyper-book* system. The objectives of the evaluation were mainly to assess system usability and utility. An evaluation conducted by assigning a number of tasks to two different groups of subjects (computer experts and computer non-experts) has demonstrated that both groups achieved a good performance; therefore, computer expertise does not represent a significant aid in order to perform better or execute tasks quickly. These results lead to the conclusion that the system is easy to learn and use by everyone, and therefore it is appropriate to employ the book metaphor.

List of Abbreviations

AAP	Association of American Publishers
ACM	Association for Computing Machinery
ASCII	American Standard Code for Information Interchange
BNF	Backus Naur Form
CALS	Computer-aided Acquisition and Logistics Support
CEC	Commission of the European Communities
CERN	European Centre for Nuclear Research
CIP	Cataloguing In Publication
DTD	Document Type Definition
EDI	Electronic Data Interchange
EMP	Electronic Manuscript Project
HCI	Human Computer Interaction
HEB	Hand-held Electronic Books
HYTIME	Hypermedia/Time-Based Structuring Language
IGBP	International Geosphere-Biosphere Programme
IR	Information Retrieval
ISBN	International Standard Book Number
ISDN	Integrated Service Digital Network
ISSN	International Standard Serial Number
MARC	MAchine-Readable Cataloguing
ODA	Open Document Architecture
OPAC	Online Public Access Catalogue
PDL	Page Description Language
SGML	Standard Generalized Markup Language
SDIF	Standard Document Interchange Format
SIMPR	Structured Information Management: Processing and Retrieval
SMDL	Standard Music Description Language
DSSSL	Document Style Semantics and Specification Language
WAIS	Wide Area Information Server
WWW	World-Wide Web

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Research Context and Objectives

The publishing process, which traditionally was based on paper as the medium of publication, is experiencing significant changes. More and more information is now published in electronic form and a number of tools already exist for handling it in this form. Electronic publishing introduces a series of new issues concerning both the strategies adopted in the publishing process and the actors involved in the publishing industry (author, publishers, libraries, bookshops, end users).

New strategies may be adopted in the process of acquisition, production and delivery of electronic documents. One of the main benefits electronic technology provides is the elimination of the editorial and production delays associated with paper documents. Other benefits are associated with the rapid delivery of information which can be guaranteed by network technologies.

Some of the actors involved in the publishing process may change their current role and their relations with others. New actors may also be necessary in this context in order to cope with those problems which are caused by the electronic nature of documents.

In the context of electronic publishing, this research concentrates on the acquisition of the source material and on the authoring processes required to produce electronic

books which provide appropriate presentation and readers' services. In other words, this research considers two main aspects related to electronic books: acquisition and authoring tools for their production, and interface design issues. The term "authoring tools" is used here to indicate a set of high level tools for facilitating the definition of an appropriate interface for electronic books, as described in more detail in the following chapters. Electronic books must be presented in a way which is suitable for a large range of potential users and does not require computer experience in order to be understood and used.

The central hypothesis of my research is that it is appropriate to translate the metaphor of conventional paper books into an electronic environment, i.e. to reproduce and extend the features and functionalities of paper books in a computerised form. The resulting electronic books are specifically designed to be parts of an electronic library. In addition, these electronic books are supposed to be produced in a semi-automatic way using a flexible publishing environment.

The use of metaphor in the study and design of information systems is already well established. A metaphor offers a helpful way of conceptualising an object or environment (McKnight et al. 1991). This approach is important as people have strong mental models of the reality they are familiar with and it is not always convenient to impose new models which are dictated by the characteristics of a new medium (Benest et al. 1987). "The term mental model is most commonly used to refer to a representation (in the head) of a physical system or software being run on a computer" (Carroll and Olson 1988).

Computer technology is evolving very quickly; very powerful and sophisticated systems have been produced. However, these systems have not always been used successfully because they can be confusing and difficult to understand. Therefore, it is important to design systems which are simple and easy to learn (Shneiderman 1992). In particular, as many users of books are not computer experts, it is important that computers do not represent an obstacle to such users.

This research has been conducted within a larger research project being administrated at the Joint Research Centre, Ispra, Italy. This project, called Super-Library, aims to provide the electronic equivalents of the features of a traditional physical library. This includes screen-based simulations of the physical organisation of the library and associated access tools, such as online catalogues, indexes of

resources, and electronic books. Several prototypes have been developed within this project both on Macintosh and UNIX environment (Catenazzi and Argentesi 1991, Gonzalez 1992, Gonzalez 1992a). The participation in a larger project has introduced certain constraints on the work described here which will be discussed in the appropriate sections.

1.2 Original Contribution

As mentioned in the previous section, this research involves two distinct activities:

- the definition of a flexible environment for publishing electronic books semi-automatically;
- the definition of an interface for electronic books which is suitable for a broad range of potential users and does not require computer experience in order to be understood and used. Such electronic books have been designed with the intention of being used within the context of an electronic library.

Significant research has already been conducted in the area, as is described in the literature review chapter. Electronic books is a topic widely discussed in the literature and there are now systems which are commercially available. Electronic publishing systems also exist; they have emerged as an evolution of the more traditional desk-top publishing systems and they aim to produce documents in electronic form rather than on paper. They are mostly intended for in-house use rather than for providing the electronic equivalent of a traditional publishing environment and its associated commercial characteristics. Some research has been also conducted in the field of electronic libraries, although this is at a more experimental stage.

The innovative aspects of this research are the analysis of the stages of development of an electronic book from the acquisition of the source material to its actual use, and the design and implementation of a system which incorporates the following features:

- a model of an electronic book based on the book metaphor and designed with the explicit intention of being part of an electronic library;
- an environment which supports the semi-automatic creation of electronic books.

Another innovative and interesting aspect of this research is the proposal of a method for evaluating the resulting electronic book. Literature concerning the evaluation of electronic books is quite poor; however, there are examples of evaluations of systems which share some similarities with my model of electronic book (hypertext and other kinds of electronic text representation).

1.3 Methodological Approach

This research has been conducted following five main steps (see figure 1.1):

• (i) the formulation of the initial *hypothesis*, a statement which determines the objectives of research. This research, as stated in the previous section, aims to demonstrate the appropriateness of maintaining the book metaphor in an electronic environment for representing electronic books.

• (ii) the acquisition of the *theoretical background* which consists of collecting, studying and analysing various kinds of material in different areas related to the research context. In particular, the areas which are considered concern different aspects related to the publishing process both on paper and electronic support;

• (iii) the formulation of a *proposal* which takes into account the initial objective and the implications derived from the previous step. This consists of designing a model for an electronic book and an environment which supplies tools for automatically generating electronic books according to the defined model;

• (iv) the development of a *prototype* which implements the design specifications defined in the previous step. In particular the prototype implements the electronic book model and provides the environment for producing electronic books;

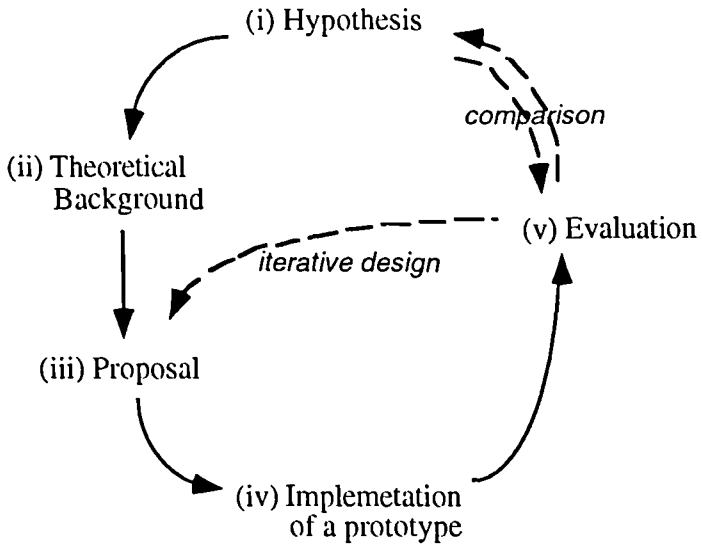


Figure 1.1: Methodological approach

• (v) finally, an *evaluation* phase, in which the prototype is evaluated. The results drawn from the evaluation are compared to the expectations expressed by the initial hypothesis. Moreover, they will be used for improving the prototype in an iterative design cycle.

1.4 Structure of the Thesis

This thesis consists of seven chapters and two appendices.

Chapter 1, i.e. the current chapter, introduces the main issues and the approach adopted in this research. First of all, the general context is presented and the objectives of this thesis are clarified within this context. The methodological approach is then described in terms of the sequence of steps which have been followed. Finally, the structure of this thesis is presented.

Chapter 2 is a literature review of a number of different topics which are all related to the general subject of this research. These topics are grouped into four main categories:

- paper books; paper books are firstly described in terms of their logical structure and format (layout); the process of paper book publishing is also described focusing on issues related to paper book design;
- electronic books; first, a comparison between paper and electronic books is proposed showing advantages and disadvantages of the two media. Then, different categories of electronic books are presented by considering several classification criteria: usage, type of information and services, kind of presentation. Design principles and techniques for producing electronic books are then discussed. Finally, some examples of electronic books are presented;
- electronic publishing; the process of electronic publishing is described in terms of the different steps involved, from the acquisition of the original manuscript to the final delivery. The importance of markup languages in the process of electronic publishing and the conversion of text into hypertext as a possible strategy for producing electronic documents are also discussed;
- the electronic library; the main features of an electronic library are illustrated, through the evolution of the paper library towards the full text electronic library. Even if it does not represent the primary focus of this research, it is discussed as it provides the context for this work.

Although each topic is presented in a separate section, there are many connections between them which are highlighted during the discussion.

Chapter 3 provides the rationale for this research. It presents in detail the focus of this work inside the publishing process, which represents the general context. Referring to the working steps mentioned earlier, the chapter represents the transition between the second and the third step: the main implications of the literature review on the formulation of the proposal are described.

Chapter 4 presents my model of an electronic book (*hyper-book*) and an environment for producing electronic books according to this model (*hyper-book builder*). *Hyper-books* are described in terms of their logical structure, their layout structure and the reader services they offer. The *hyper-book* model is also formally described as a dynamic system. The *hyper-book builder* provides the environment for converting electronic marked texts into *hyper-books* in a semi-automatic way. The conversion process and the editor tools which are available in this environment are described in detail. The potential of *hyper-books* in the context of the electronic library is then illustrated. Finally, the *hyper-book* model is compared to the paper book at different levels: task, semantic and lexical. The last level introduces a number of features which are fully described in chapter 5.

Chapter 5 describes the implementation issues related to the development of the prototype. The chapter organisation mostly reflects the structure of the previous chapter, although the same concepts are presented from a practical perspective. The chapter is divided into two main sections, one is related to the *hyper-book* implementation, the other to the *hyper-book builder* implementation. Finally, the comparison between *hyper-books* and paper books, proposed in the previous chapter, is completed by considering them at a physical level.

Chapter 6 describes how the *hyper-book* evaluation has been conducted and the results drawn from the evaluation. First of all, a brief introduction to the main concepts and techniques in the evaluation field is proposed. Secondly, the objectives of the evaluation and the test application are described. Thirdly, the procedure followed during the evaluation is outlined. Finally, the results are reported in the form of diagrams and tables and the main conclusions drawn from the evaluation are then presented.

Chapter 7 summarises succinctly the conclusions and lessons learned from the work carried out and proposes a number of possible improvements and further work. First of all, general conclusions are presented, then the conclusions related to the *hyper-book* and those related to the *hyper-book builder* are outlined.

Appendix A contains the *hyper-book DTD*, i.e. a document type definition which specifies the logical structure of a class of documents, scientific books. The electronic documents used as input during the *hyper-book* generation process are marked according to the *hyper-book DTD*.

Appendix B presents the questionnaire given to the users and the sequence of tasks assigned during the evaluation.

Appendix C reports in details the users' answers to the questionnaire and the results of the execution of the tasks, in form of tables.

Chapter 2:
The Literature Review

2.1 Introduction

This chapter discusses a number of topics which are connected to the production and presentation of books, both in paper and electronic form.

Whereas there are well established conventions for paper books, which are the result of a long process developed over the years, electronic books are a relatively young concept, which needs to be further explored, as the technology evolves.

Paper books exist in a variety of types and structures: the type of book determines in some way the underlying structure. For instance, fiction books usually have a linear structure; a manual mainly has a hierarchical structure; a dictionary is organised essentially as a hypertext (i.e. as a network of interconnected nodes of information). In spite of the different structures of books, the medium used for presenting them has a strong influence. While the paper medium presents information in a sequential or linear way, the electronic medium is much more flexible and dynamic and allows non-linear documents to be presented in a way that maintains their non-linear nature. Linearity can be seen as an intrinsic property of the written language (McKnight et al. 1991), but not of the message that it conveys. A non-linear document can benefit from being presented in electronic form.

The advantages provided by the electronic medium in designing electronic books are widely discussed later in this chapter.

Similar considerations can be made for the book publishing process: whereas the paper publishing is now a well established process, electronic publishing is a relatively new process and there is a lot of research about it. The potential of electronic technology is enormous, however a set of new issues and problems have emerged. This is also a subject widely discussed in this chapter.

Electronic publishing has implications on the traditional channels of document delivery and on the infrastructures traditionally used for storing and making books accessible. Libraries are being considerably affected by the electronic revolution.

This chapter contains a series of literature reviews covering the topics discussed above, including:

- the concepts of the book in both paper and electronic form. One of the objectives of this review is to deduce design principles for electronic books both from traditional book design and from existing studies and experiments with electronic books.
- the concept of publishing; the focus is on the electronic publishing process. The main steps involved in the electronic publishing process are outlined from the acquisition of the original text to the final delivery. Markup languages have a fundamental role in many aspects of electronic publishing and they are described in a separate section;
- the concept of the electronic library; even though it does not represent the primary focus of this research, it has been discussed where it provides context for this work.

2.2 Paper Books

The most important event in the history of the book was Gutenberg's invention of the printing of movable characters in the 15th century. Printed books replaced the traditional manuscripts. The number of available books rapidly increased. The book became an object with specific features and design rules, something very close to the modern book. Print led to pagination, indices and bibliographies; print increased literacy, democratised knowledge, and created public libraries. New professions and activities were created around the printed book such as the designer, the publisher and the book seller. At the same time the author assumed a more important role and the book market started to develop together with rules covering copyright. The book thus became an object with deep historical value and a classical tool for information delivery.

The modern book has acquired a number of features, both in structure and format, which are now universally recognised. They will be discussed in this section.

2.2.1 The Structure of the Book

The term structure is used to indicate the logical components of a book. A traditional book may be thought of as consisting of three main parts (Martin 1989, Haines 1965):

- the preliminary pages;
- the body of the text;
- the end pages;

The preliminary pages include the title page, the copyright page, dedication, the Table of Contents, the list of illustrations, acknowledgements, foreword, preface and introduction. The title page represents a special highly condensed message for the receiver. Objects are emphasised in this part in order to be taken at a glance. The title page normally contains the title, the author's name, the publisher's name, and usually the year of publication. The isolation of the author and the title at the head of the page is a convention.

The verso of the title page, biblio or imprint page, contains bibliographical details of the specific edition: the copyright year, the printer's imprint, etc. The Table of Contents outlines the organisation of the book and indicates the material contained in it. In other words, it is a map of the book. The list of illustrations should follow the Table of Contents. Acknowledgements usually appear in the preliminary pages, but they

could appear at the end. The foreword is a commendatory note from an illustrious hand (Martin 1989). A preface concerns how the book came to be written. It normally appears after the foreword and before the introduction. The introduction is a statement of themes, scope, structure and conventions. It is the opening chapter of the book. Introduction, preface and foreword are usually set to the style of the text.

The body of the text contains the main corpus of the book; it is hierarchically organised: it consists of a sequence of chapters which are organised in sections and subsections.

The end pages include appendices, notes, glossary, bibliographies and index. They are useful accessories and reinforce the book content. Appendices contain additional specifications that may be useful to consult. Notes can appear as a component of the end pages of the book, but more frequently they appear on the same page where the note is referred to, and not very commonly at the end of a chapter. The glossary can be viewed as being a mini-dictionary. The bibliography is particularly important in scholarly and scientific books. There are rules for bibliographic references, although there are many different conventions. The index is an indispensable part of the book, as it provides a form of keyword access to the book content. The value of the book can depend significantly on the index it provides.

The above is a general description, which is appropriate for scientific books which are of particular interest for this research. However, not all books follow this arrangement: in some there may be additional components; in others, such as books of fiction, not all these components appear.

2.2.2 The Format of the Book

An interesting perspective on the configuration or format of a document is reported in (Southall 1992). Such a configuration depends on rules of presentation and rules of composition. The rules of presentation, associated with the document, are organised hierarchically. They dictate the size, the position of the different blocks in a document, the set of fonts used, line spacing, justification, page breaks, etc. The rules of composition, associated with the formatting system itself, include spacing rules which determine space between character in a word and space between words. This section focuses mainly on the presentation rules.

The format of a book is a term which is commonly applied to the general make up of the book. It concerns aspects such as the size, the typeface, the page, and the margins,

which will be described in the rest of this section. Format differs from book to book according to the type of the book (novel, textbook, reference book, etc.) and the publishing house style.

The size

One of the main aspects that influences the format of the book is the size. Two parameters are involved: thickness and height. It is important to maintain a good proportion between them. "Any book that has to be read with comfort should be small enough to hold conveniently and large enough to permit a size of type that is easily read" (Haines 1965). There are various book sizes, indicated by terms such as folio, quarto, octavo, twelvemo, sixtemno. These sizes are historically derived by the folding of the printed sheets. Large sheets of papers used in printing were manufactured in standard sizes. For library handling and for reading use, it is better to have medium sized books (ranging from six and a half to nine inches).

Typefaces

The typeface is another important aspect of the book format. Typeface may be defined as "a distinctive, visually consistent design for the symbols of an alphabet" (Rubinstein 1988). Examples of typefaces include Palatino, Baskerville, and Times. The term *typeface* is sometimes erroneously confused with the term *font*, which indicates "a fixed example or instance of a typeface, usually of a single size and variation, and often tailored to a particular output device" (Rubinstein 1988).

Selecting a typeface involves thinking about factors such as the book type, the book usage, the size of the page, the fact that a book may belong to a series, the presence of illustration, etc.

An old and still valid suggestion in choosing a typeface is to use one typeface for one book with different sizes and styles for headings and emphasis on display. Continuous reading is highly degraded by the introduction of special styles for some words. A text should be clean, and well-ordered in order to not disturb the concentration of the reader. "Over emphatic headings can result in making it difficult to follow the structure of a book" (Martin 1989). In particular, bold should not be used in continuous reading. Rubinstein (Rubinstein 1988) suggests using italic for inline textual emphasis and bold for greater emphasis such as headings.

There are some display sizes of text faces that are normally used. A range of text sizes from 5pt to 8pt are used for notes, appendices, indexes. From 9pt to 14pt for continuous reading. From 14pt to 24pt for chapter and major sub-headings. From 24pt

to 48pt for main lines on title pages. When an increase or decrease of the text size is required, it is convenient to use a step of two points.

Other considerations about typefaces deals with legibility. Legibility can be defined as the degree to which text is easy to read (Rubinstein 1988). The space and the leading of the page represent important factors. The leading between lines should not be so large that white space dominates the stream of print, nor so reduced that the page seems a solid mass of print. Line widths, leading and type size must be chosen properly.

The page

The page is another essential aspect. Book designers often use a grid as an aid to the layout. "The grid is a rectilinear subdivision of the page that provides a template for the layout of each page ... The underlying structure provided by the grid is the basis for the reader's conceptual model of the structure of the page" (Rubinstein 1988).

Many objects contribute to create a page: margins, header, footer, text, illustrations, and tables.

Special consideration has to be given to the margin size. Wide margins may facilitate the reading process by helping the reader avoid distractions from objects near the text and by preventing a crowded look on the page. Margins are indispensable in order not to cut off part of the text when the book is trimmed. Margins also represent a place where annotations and comments can be written. Another good reason for margins on paper is that they keep the printed part of the book flat (Rubinstein 1988). For most books, the main body occupies only 50% or 60% of the page. A relation exists between the different margins. The inner margin is the narrowest (more or less half the width of the outer). Then there is the top margin, the outer and the bottom.

Running headings are also important elements of a page; they provide a means of location and navigation through the book. Careful positioning is essential. Usually the title of the book is placed on the left-end page and the title of the chapter on the right-end page. Chapter titles may be too long and require shortening. The running headings do not appear on the opening page of any chapter nor on preliminary pages. The footer usually contains page numbers, which may also be placed at the top of the page.

Other considerations deal with the text. Several different typographical aspects are associated to a block of text; they deal with typeface discussed above, line width, interline space and justification of lines. Illustrations and tables appear embedded in the text, possibly close to the text which refers to them.

The appearance of the page is also related to how the process of breaking text into pages operates, which is the subject of the next section.

2.2.3 Pagination

Pagination is the process of laying out the parts of a large document into pages. (Rubinstein 1988). Text cannot be divided into pages blindly. A number of typographical constraints and goals have to be satisfied (Martin 1989, Rubinstein 1988):

- general requirements imposed by the documents style have to be taken into account;
- widows and orphans must be avoided. A widow is an isolated line set at the top of a new page, usually it is the last line of the paragraph on the previous page; an orphan is a single line left alone at the bottom of a page. Their presence may confuse while reading;
- irregularities in spacing attract the reader's attention and reduce the quality of text presentation. Extra space within words can degrade the legibility of print. Space between words must not exceed the space between lines; this is known as *river situation.*;
- a line of space should not be used as an alternative to proper indentation;
- hyphens at successive line endings can halt the reader. In addition, a word must not be hyphenated across a page break;
- the facing page should have the same depth.

Complex documents, i.e. documents including footnotes, marginal notes, illustrations, and tables, create several problems during the pagination process. Each element needs a specific treatment:

- footnotes are difficult to position. Ideally they should appear on the same page as their mark;
- a similar issue is that of marginal notes. They should be placed entirely on the page to which they refer. The problem is that their reference point can be toward the bottom of the page and, in this case, it is difficult to insert the note completely on that page;
- illustrations and tables should appear near the text where they are referenced;
- cross-references which contain page numbers must be modified every time a change occurs as a consequence of a repagination.

Although several constraints have to be taken into account during the pagination process, there are also many degrees of freedom that allow different arrangements of

text and illustrations. The pagination process is usually handled by automatic layout systems (desktop- and professional-publishing systems). Finding the uniquely best solution to paginating a document is computationally very expensive, even if heuristic methods can help. Therefore users must interact with the system to make further arrangements.

2.2.4 The Paper Publishing Process

The main players involved in the traditional publishing process are:

- the author, who writes manuscripts to submit for publication;
- the editor, who is responsible for reading, correcting and evaluating the author's manuscript;
- the designer, who is responsible for defining the typographical aspects of the book;
- the typographer, who actually implements the designer's directives and produces the paper book;
- the publisher, who publishes the final book, i.e. he makes the book available to the readers.

The role which is most interesting for the purpose of this research is the designer role. "Design is the sum of the decisions which make a product serviceable and attractive" (Martin 1989). The book designer is a visually literate person; he is responsible for the visual presentation of the author's manuscript to the readers.

The process of book preparation requires the understanding and application of concepts developed over years in craft typography. "Good typography is to be a silent art" (Rubinstein 1988): the goal of typography is to communicate, not to call attention to itself as the vehicle. Typographical rules typically determine operations such as line-breaking, and grid layout, as has been discussed in the previous sections. Few typographical rules are universal: factors like the purpose of the document, the cultural background, and the school of typography determine the rules to be applied.

Nowadays, computer technology is applied to the process of preparing books. Digital typography (also named automatic typesetting) is the technology of using computers for the design, preparation and presentation of paper documents (Rubinstein 1988, Mittelbach and Rowley 1992).

The process of designing, preparing and presenting electronic documents is another interesting subject of investigation, which will be discussed in the next sections.

2.3 Electronic Books

Electronic books are made possible by several technological improvements (powerful processors, high density storage, etc.), a significant decrease in the hardware costs, and interchange standards for making information accessible (Reynolds and Derose 1992). Desktop publishing systems, which have principally focused on paper output, represent in some way the transition from paper to electronic books. New systems are now starting to appear, which are specifically designed for electronic output. Paper is becoming just one of several media used to distribute information.

By examining the paper book it is possible to identify a number of advantageous and disadvantageous characteristics, which essentially depend on their static nature. Positive features include (Yankelovich et al. 1985):

- historical value (well established design and typographic conventions);
- access without extra equipment;
- portability;
- possibility of browsing;
- possibility of annotations and underlining;
- a well defined method for reproduction;
- high resolution print and graphics;
- ease of reading;

On the other hand there are many limitations associated to the paper books which are due to the *static* and *non-reactive* nature of the paper support (Barker 1992, (Yankelovich et al. 1985)):

- difficulty of updating the content;
- difficulty of customising information;
- difficulty of locating information;
- absence of sound, animation and moving pictures;
- cost of dissemination;
- difficulty of reproduction;
- possibility of being easily damaged.

Electronic books, allow these limitations to be overcome thanks to the facilities offered by an electronic medium, namely flexibility and interactivity. A general

definition of electronic books is given in (Barker 1993): electronic books are information delivery systems that are capable of providing their users with access to pages of reactive electronic information with which they can interact. These pages are conceptually organised like the pages of a paper book.

Electronic books provide a number of advantages:

- they can react and respond to their user in a dynamic and flexible way;
- they can be dynamically changed according to user needs;
- they can provide history mechanisms, for keeping track of user interaction with the book;
- they allow non-linear paths to be followed while using the book;
- they can provide searching facilities for effective retrieval of information;
- they can supply more potential channels of communication compared with conventional books; multimedia facilities (sound, moving images, pictures) all can contribute to create a richer information environment and can be incorporated into electronic publications with relative ease.

In order to make these facilities really effective and usable, an important requirement is that electronic material is presented in a familiar and user-friendly way, thereby limiting the cognitive overhead associated with having the computer as an intermediary between readers and books.

Many approaches are possible in order to guarantee increased usability with an electronic medium: the use of online help; the use of menus instead of, or in addition to, a command language; natural language interfaces, the use of metaphors, etc. The approach that will be explored in this paper is the last one, i.e. the emulation of real objects in an electronic environment. In particular the metaphor of a book will be considered.

2.3.1 The Book Metaphor

Metaphors have been widely used to facilitate learning of computer systems. A model with which the user is familiar is provided in order to enable users to exploit their existing knowledge of other domains when learning to use a system (Benyon et al. 1990). Metaphor theories and their use for interface design are well-documented in the literature (Carroll and Mack 1985, Carroll et. al. 1988, Carroll and Olson 1988, Hammond and Allinson 1987).

Hammond and Allinson (Hammond and Allinson 1987) stress the importance of integrating the metaphor in the design of the system: a metaphor can serve the dual

purposes of disciplining the designer and helping the user to learn. They argue that there are two relevant dimensions for understanding the information which metaphors convey: scope and level of description.

Scope refers to the number of concepts addressed in the user's understanding of the system. Metaphors of wide scope attempt to encompass all or much of the functionality of the system and the tasks it supports. Examples include the metaphor of the desk-top and the book metaphor. A more narrow scope metaphor is the hydraulic metaphor, involving pipes and tee-pieces, in the Unix system.

The metaphor's level of description refers to the type of knowledge it is intended to convey. This may be very high level information such as how to think about the tasks and their completion, or very low, such as how to think about particular command syntax in order to remember it easily. Hammond and Allinson (Hammond and Allinson 1987) consider four different levels: task, semantic, lexical and physical.

The *task* level concerns the structure and constraints which the system imposes on potential user tasks; it would help answer the question: "can I achieve goal A?"

The *semantic* level defines the facilities provided by the system in order to accomplish the tasks mentioned above; it would help answer the question: "what does facility B do?"

The *lexical* level defines the terminology or language used for referring to the system entities and operators; it would help answer the question: "what does the term C mean"?

The *physical* level specifies the physical representation of the interface and the actions performed by the user; it would help answer the question: "what activities are needed to achieve D?"

Metaphors do not need to be appropriate at all levels of description. Metaphors do not need to explain everything of the system and not all features of the metaphor have to map to the system. If the metaphor seems to be inappropriate at some level of description, it is better not to force it. An important rule is to avoid erroneous mapping between the metaphor and the system domain, because the user may be misled.

Typical examples of metaphors which have been employed in interface design include the desktop metaphor, used for an operating environment, the typewriter metaphor used for word processors, and the book metaphor, used for electronic documents representation. Metaphors may also be combined: for instance typical metaphors which are used in hypermedia systems include the story, guide, museum and book metaphor (Barker 1993).

The metaphor which appears to be the most suitable for presenting electronic books is the paper book metaphor. There are several authors who support the idea of preserving the book metaphor in electronic books. Reynolds (Reynolds and Derose 1992) considers the interface design crucial to the success of electronic book projects and proposes an interface which offers a paper-like feel and overcomes the paper book limitations. Barker starts from the premise that everyone is familiar with the nature and use of books (Barker 1991) and exploits the book metaphor in designing electronic books. Benest (Benest *et al.* 1987) agrees with this position and states that "Interactive mechanisms must interfere as little as possible with the task that the human is actually performing". How people look for and read a book and the recognition of the type of the book (novels, text books, paperbacks, etc.) are subconscious processes. The behaviour is influenced by the degree of familiarity people have with the book and by the kind of book. The conclusion is the same: it is important to maintain on the screen a similarity with the paper counterparts (Burrill and Ogden 1989).

An electronic book should offer at least the same access facilities as a paper book (Tables of Contents, back of book index, turning pages, the possibility to jump to another part, etc.). Further, existing facilities can be extended. For instance, a Table of Contents can be presented as an active list of headings; it is usually dynamic; it can easily be expanded and closed. Other solutions for the content page are possible in a computerised book which depend on the application domain. For instance, in a geographic book there is a strong location element to each heading and it is convenient to display the sections representing a location according to their geographical position.

Special consideration has to be given to personalisation facilities. In a real book there are many forms of personalisation (Burrill and Ogden 1989): the most common is *marginalia* (parts of the text are marked in some way). This can be realised in the computerised book by allowing readers to make comments in the margins or to highlight sections of text. Another form of personalisation, *association*, indicates how frequently a page has been accessed. "Each time a page is read a few more random pixels are drawn onto that page in order to give it well-thumbed look". A third form of personalisation is *actuality*, i.e. the equivalent of bookmarks.

2.3.2 Classifying Electronic Books

There are different ways of classifying electronic books according to different criteria. Collis (Collis 1991) classifies electronic books as reference books and text books. This distinction is based on different usage of the book.

Barker (Barker and Giller 1992) defines a simple taxonomy based upon the medium that electronic books are published on, the functions they perform and the type of information and services that they provide, and classifies electronic books into four classes: archival, informational, instructional and interrogational. Barker (Barker 1992, Barker 1993) also proposes another more fine-grained taxonomy based on two criteria: the type of information and the kinds of services they are able to provide. The categories are:

- text books, which are composed entirely of textual material;
- static picture books, which are composed of pages which contain one or more pictures;
- moving picture books, which contain mainly moving imagery;
- talking books, which use recorded sound in conjunction with a variety of interactive audio techniques;
- multimedia books, which can be created from any combination of text, static picture, animation and sound, and are delivered by means of a single delivery medium;
- polymedia books, which use a combination of different media (CD-ROM, paper, magnetic disc, etc.);
- hypermedia books, which are like multimedia books with the peculiar feature of a non-linear organisation of information;
- intelligent electronic books, which incorporate artificial intelligence techniques in order to provide advanced reader services;
- telemedia books, which attempt to use the benefits both of CD-ROM based multimedia books and broad band electronic communication networks (multimedia e-mail, tele-conferencing, remote access to distributed sources of information);
- cyberspace books, which provide access to various types of virtual reality facilities.

This classification moves from very simple books towards more sophisticated and powerful books, which are made possible by exploiting the most innovative technological tools. One outcome of this enrichment is that the book metaphor could be distorted. The addition of services and the inclusion of media which usually are extraneous to a real book, must be considered very carefully in order not to obtain an object that cannot be recognised as a book by the user.

Another important criterion for classifying electronic books is based on the kind of presentation. Different authors provide different interpretations of the concept of electronic book, although they are all based on the idea of using the paper book as a

model for the electronic one. These categories are not necessarily exclusive, but may overlap.

First is a large group of electronic books, called *portable electronic books*, whose main purpose is to reproduce portability (Feldman 1990, Lande 1991). Clearly, two of the most appealing features of paper books are the facts that they can be taken everywhere and read without the need for other external resources other than light. The electronic equivalent has many drawbacks: limited screen size, limited resolution, etc. To date, portable electronic books have made their most significant impact in the area of linguistic publications: dictionaries, thesauruses, spelling checkers, etc. Feldman (Feldman 1990) defines these portable books as HEB, Hand Held Electronic Books, and identifies two other sub-categories: the dedicated HEB, which offers a fixed data content, and the generic HEB, less diffused, which is more like a reader device into which different data modules can be plugged.

A second group of electronic books is more concerned with preserving the logical structure rather than the physical nature of the book: information is usually displayed in scrolling windows, which may overlap. The idea of the page, as a static block of text, is not considered. This is the case with Dynatext (Electronic Book Technologies 1990) and SuperBook (Egan et al. 1991). Both of these provide full-text indexing, links, navigation and orientation through a dynamic table of contents, and a multi-window text display. An interesting aspect of these two systems is the fact that they provide the capability of automatically importing text which is electronically available in different format.

Finally, there is a group of electronic books which support both the physical and logical aspects of the book. This, in the authors' opinion, is the most complete and satisfactory solution because both logical structures and physical clues (e.g. pages, book thickness) are indispensable for recognising an object as a book. This approach becomes particularly important when a paper version exists of the same document and readers are familiar with it. Section 2.3.4 presents several examples of electronic books which mostly belong to this category.

2.3.3 Design Issues

The communication of knowledge is a basic requirement for all types of book, and particularly for electronic books where methods of information and knowledge delivery are different from those used in conventional paper books. Essential to electronic book design is an understanding of user profiles and of the purpose of the book, which determine the choice of interaction strategies and types of interface.

Barker (Barker 1991, Barker 1992, Barker 1993) analyses in detail the problem of designing electronic books and proposes a set of high-level design models and a number of pragmatic guidelines.

Models include a *conceptual model*, intended for end-users as an orientation tool, which presents an electronic book as composed of pages of reactive and dynamic information together with a number of book control functions. The other models, the design and fabrication models, are intended for designers and producers. The *design model* involves a number of choices dealing with the types of end-user interface, the styles of interaction, the facilities and services provided to the users, the control mechanisms, the overall book structure, the page structure, the content of the book and the media mix to be used. An element which can facilitate the creation of electronic books is the provision of a number of templates, which are pre-formatted empty book structures and empty pages which may be filled by the authors with information appropriate for a particular application (Barker and Manji 1991). The *fabrication model* includes all the development stages from the initial design phase through the prototyping to the final production stage.

A set of guidelines for electronic book design are proposed which fall into six basic categories: knowledge engineering, page design, interaction style, end-user tools and services, multimedia and hypermedia. For instance, referring to knowledge engineering (i.e. the task of creating the underlying knowledge corpus), it is important that this knowledge is built correctly in order to have a good resulting book.

The design strategies used for paper books provide possible directions for electronic book design, while recognising that its different nature makes it necessary to develop specific procedures. The electronic book design is conditioned by electronic support constraints (limited screen size, resolution, etc.) and facilities (flexibility, interactivity, etc.).

The description of paper book components proposed in section 2.2.1 could also be valid for electronic books. However, while conventional books have generally a strictly linear organisation of pages, electronic books can be organised in non-linear structures. Many considerations about the format of the paper book represent a good basis for electronic book design, but cannot be adopted entirely. The size, for instance, is constrained by the size of the screen. The font size should be bigger on screen than on paper in order to be legible.

The page in electronic books can be a dynamic object; it can be created when it is needed and presented to meet a user's needs. Moreover, it is possible to introduce new

types of structural entity such as subpages and page clusters. The page of an electronic book has to support the information display function of the conventional book, as well as the book control function that is a proper feature of the electronic book (Barker 1991). A number of operations are executed by generic control functions, such close book, next page, previous page, go to page N. Other control functions may exist, which are designed specifically for the underlying application. For instance, an electronic calculator may be necessary in mathematics books.

Margins should also be incorporated. Many of the reasons which justify the presence of margins on paper are no longer valid for screens. Screen space is more precious than paper and many screens do not provide enough room for margins. However, it is convenient to maintain them both for aesthetic and historical reasons, as paper books are conventionally printed with margins (Rubinstein 1988).

2.3.4 Some Examples of Electronic Books

There are many projects that appear very relevant to the design of electronic books. They are all based on the idea of using the paper book as a model for the electronic one, because people are familiar with that model. The best way to do this is to determine which characteristics of the paper book people use and feel comfortable with, and explore ways in which these characteristics may be realised on the screen.

This is the philosophy behind Vortext (Burrill and Ogden 1989). Vortext is a valuable example of a computerised book because it presents many innovative features. The most original idea of this project deals with how to approach the problem of providing orientation facilities within the book. In a real book they are provided by the use of running headings and the thickness of the paper itself. The solution adopted in Vortext is the idea of closed pages. The text is displayed between two sets of vertical lines which represent the thickness of pages before and after the current page. Section names are indicated between these lines and may be selected in order to go directly to that page. Another problem that is addressed in Vortext is the personalisation of a book (see section 2.3.1). Vortext implements actuality and gives ideas for implementing the other two forms: *marginalia* and *association*. A bookmark (*actuality*) can be inserted into the current page of the book by dragging the pointing device in order to turn down the corner of the page. Other aspects which have been considered include the content page which is presented as a dynamic table of contents, and the index; the problem of both creating and representing an index has been addressed in Vortext.

Benest (Benest et al. 1987, Benest and Duric' 1990) presents an example of the electronic book within more general contexts: the electronic library and the automated office. The environment is designed so that the metaphors of book, library and office are respected. When a book is selected in the bookshelf, it is presented like an open book. "Pages may be turned using mouse keys. By holding down these keys, the user may flick pages. The black band along the side and the bottom of the book provides the user with an implicit cue of his location within the information. Selecting with the black band enables the user to make large jumps." Books have a consistent structure: a Table of Contents in the front, an index and references at the back. In addition, active areas can provide non-linear progression through the text. The text page may be annotated and highlighted. An eraser allows removal of both of them. Bookmarks may be inserted. If a reference is selected in the text, the full reference is shown nearby. Selecting a reference within the list of references causes a list of page numbers to be displayed where the reference occurs. The personalisation facilities are very well designed. Benest's book is assessed by McKnight (McKnight et al. 1989) as one of the most interesting approaches in providing metaphors as navigation support. However, McKnight, recognising its validity, criticises the degree of simulation of the paper counterpart. He argues that not all types of texts need to retain such detailed aspects of the paper book.

Another system based on the book metaphor is the OpenBook hypermedia system (Ichimura et al. 1993). The system allows a user to leaf through a set of nodes just like a book. This type of browsing allows a user to evaluate rapidly a large amount of material and to determine what is useful. Leafing through an electronic book takes advantage of a cognitive capability to skim or perceive the outline of the contents. Moreover, the system supports a query-based access mechanism to facilitate global data access.

Other valuable examples of electronic books have been developed by Barker and his group within an "electronic book" project (Barker 1991, Barker 1992, Barker 1993). They include, for example, the ABC book for early learners (Barker and Giller 1991), an interactive package for teaching people good screen design practice for Computer-Based Training, and a Static Picture Book with Audio Narrations. The common aspect of these books is that they are all *instructional* books, i.e. books which support learning and training activities. The experience gained from the design, production and evaluation of these books, and the study of a range of commercially available publications led to the definition of a number of high-level design models and a

collection of pragmatic guidelines for producing electronic books (as indicated in the previous section).

Blake (Blake 1988) proposes a system, Smart Book, which is an example of generic HEB publishing. It consists of two components: a reader unit and a book pack. The functionality of the book is unique to the book and is provided by the program contained on the book pack. The reader unit contains generic operations for those functions common to all books (page turning). Problems such as the interface design are not well considered; the description is more oriented towards technological aspects. There are several examples of dedicated HEBs (Feldman 1990) now commercially available: the Spelling Ace, the Pocket WordMaster, the electronic edition of the Bible produced by Flanklin, WordFactory produced by Microlytics, etc.

At present, there are several other examples of commercially available electronic books. One of these is the Expanded Book Toolkit, developed and commercialised by the Voyager Company (Rosenthal 1992). The system brings together the benefits of the printed page and the power of computerised text-handling, word searching, and annotating. In addition, it provides the tools for creating other Expanded Books by converting text from a range of word processing and page layout files.

Other examples of commercially available electronic books, most of which are available on CD-ROM, are reported in (Barker 1992, Barker 1993): the Grolier Encyclopaedia, the Compton's Multimedia Encyclopaedia, the Discis Talking Books, the Microsoft Bookshelf, the Sony Data Discman publications including the Concise Oxford Dictionary and Thesaurus, Hutchinson's Encyclopaedia Dictionary, etc.

In summary, most of the systems described above have originated from the idea of using the paper book metaphor for electronic books. Benest's book seems to be the closest one to this idea: the book is presented just like an open book, active back bands for simulating thickness, and personalisation facilities are all reproduced as in the paper model. Vortext approaches the same problems in a different way though some solutions seem to be not so natural and intuitive. Barker's electronic books (Barker and Giller 1991, Barker 1991, Barker 1992) are based on the paper book model and provide specific properties because of their instructional nature. Finally, portable electronic books and electronic books available on CD-ROM represent the categories mostly in the market.

2.4 Electronic Publishing

Raitt defines electronic publishing as the preparation, storage and dissemination of information using computers, telecommunications and terminals (Raitt 1985). The availability of new technologies, e.g. enormous storage capacity, processing speed of computers, and communication capabilities, strongly affects the electronic publishing development.

A number of issues arise as a consequence of this development. The ownership of the results of computer manipulation is an important issue. Who is the author of a product that can be easily created from existing pieces of material? Will the publisher be able to keep the classical role of gatekeeper, i.e. his authority to decide which works to publish? How is it possible to control user access to information? The rapid development of electronic publishing also has impact on copyright. Copyright gives creators and publishers the tools to protect their intellectual property and their financial interests. Electronic publishing requires some adjustments in the copyright model, as it provides a number of new capabilities which are not available in the paper version (ease of replication and transmission, ease of modification, etc.).

There are many points of view about the possible consequences of electronic publishing on the traditional publishing industry. Lancaster's prediction (Lancaster et al. 1979) that print-on-paper material would be replaced by electronic publishing and libraries would disappear by the year 2000, is certainly not going to come true. Most authors (e.g. Wang 1986, Jackson 1992) agree on the fact that printed material will not be replaced, because of its portability, convenience, durability, low cost and human reading habits. They believe that electronic publishing might become a supplement rather than a replacement for traditional communications media. This forecast has been demonstrated to be true at the present time. Paper has not shown any signs of declining in importance as a medium; paperwork has actually increased as a direct consequence of the electronic technologies (e.g. printers). Recent trends show that it will be the same in the future. The society of the future will consume more and more information in both paper and electronic form. Feldman (Feldman 1990) agrees with this point of view stating that, even if the printed word increasingly becomes one option for delivering information, it will continue to flourish where it has the greatest utility for the information users.

This section first compares electronic publishing with traditional paper publishing; then it describes the process of electronic publishing in more detail distinguishing between the different steps involved. Finally it discusses the importance of markup languages in the process of electronic publishing and the conversion of text into hypertext as a possible strategy for producing electronic documents.

2.4.1 Comparison between Electronic and Paper Publishing

Electronic publishing offers a number of advantages over traditional paper publishing.

One of the main benefits (Dodd 1990) electronic technology provides is the *elimination of the printing delays* associated with paper document production, i.e. the time spent on the editorial and production phases. This is due to certain changes within the technological environment: the widespread use of wordprocessors, the increasing use of floppy disks as an exchange medium, electronic mail facilities, and high quality laser printers. These changes affect the production of journals more than books, as journals require a shorter turnaround time.

Other advantages are relevant to the post publication period. One of these is the *rapid delivery* of information that can be guaranteed by network technologies. Another one is the possibility for users to select parts of a document they are interested in and get only the selected parts (*selective publishing*). In addition, it is possible for the user to define *an interest profile*, so that the user is automatically informed when new, highly pertinent documents match this profile. Negroponte's idea of idiosyncratic systems goes beyond this: "he suggested that electronic documents adaptively display not only what material readers wish to see, but also how they would like it presented" (Yankelovich et al. 1985). Electronic documents are *dynamic* and *flexible*: both authors and readers can customise the material (Yankelovich et al. 1985). Updating electronic documents is therefore less expensive and time-consuming than paper documents. *Navigation supports* can be provided through the use of hypertext structures. Another advantage of electronic documents is the ability to include a *variety of media* (static images, but also animation, videos, sound, etc.) which add a richness to the electronic document systems impossible to recreate with paper media. "The *environment advantages* of using less paper is also significant" (Reynolds and Derose 1992).

The next table summarises the main features of Electronic Publishing.

	ADVANTAGES	DISADVANTAGES
Production and delivery	- fast - selective publishing - interest profiling - environmental adv.	- not well established methods
Document properties	- ease of updating and customising - navigation support - multimedia facilities	- low screen resolution

Table 2.1: Electronic publishing

On the other hand, paper publishing presents many advantages over electronic publishing.

Production and dissemination of paper products are governed by processes which have long been established. It is not the same for electronic publishing, which is a still in an infancy phase.

Other advantages are related to the *ease of access*: paper is portable and no additional equipment is required to access the information (Paulapuro 1991). Paper documents have a concrete form (Paulapuro 1991) and provide a *definitive version.* Hard copy makes people comfortable with the information, makes them believe that nobody can change it. Lie (Lie and Bender 1992) says that the physical properties of the paper media represents a big obstacle for computer display to compete with paper. The drawback of providing a definitive version is that no selective publishing is possible with paper material: a paper document has to be delivered in its complete form.

Another important factor to consider is the *reading effort.* Although a large number of studies claim that reading from screen is slower than reading from paper (Rubinstein 1988, Gould 1987), there are other studies which demonstrate that there is not an absolute superiority of one on the other medium (Hansen and Haas 1988, Gould et al. 1987). McKnight (McKnight et al. 1991 and McKnight et al. 1989) compares reading from screen vs. reading from paper focusing on human factors more than on the technological aspects. His discussion takes into account several experiments already

conducted in this area. Five potential differences are highlighted: speed, accuracy, fatigue, comprehension and preference. Even if many experiments reveal a better performance of paper in most of the cases, McKnight argues that much of this work was carried out on poorer screens than are currently available; the technological improvements should guarantee an image quality on screen comparable to the paper quality.

Table 2.2 summarises the main features of the paper publishing.

	ADVANTAGES	DISADVANTAGES
Production and delivery	- well established methods	- slow - all or nothing
Document properties	- portable - no need for extra equipment - good readability - tangibility - definitive version	- static and non-flexible

Table 2.2: Paper publishing

Another point where paper and electronic publishing differ is related to *economic aspects* such as the use of search facilities, distribution expenses, etc. Whether this is an advantage or not depends on the viewpoint. From the reader's point of view, the possibility of having material of direct interest using search facilities is an advantage. He pays only for what he reads. The distribution of electronic documents on demand saves considerable costs associated to printing, warehousing, transportation and additional costs related to overestimating or overestimating demand (Rawlins 1993). However, the initial investment for building an adequate infrastructure for delivering and presenting electronic material is also relevant. "There would be added economic complexities as the current distribution services would be replaced by network providers, the printing industry by online database services" (Dodd 1990).

2.4.2 The Electronic Publishing Process

The process of publishing electronic documents consists of three main steps:
- acquisition of the raw material;
- processing of the material;
- delivery, i.e. distribution of the material to the users.

• Acquisition

The acquisition process is largely affected by the support on which the material is originally available (Lu 1993). If only a paper version exists, documents need first to be converted into electronic form. This is usually accomplished using a scanner. Most of the time, OCR is then applied to recover the text in machine readable form. The process of character recognition is not simple; it is dependent on factors such as the clarity of type and the presence of diagrams and pictures. Further checking and verification is always required.

When documents are already available in electronic form, the problem is easier. A distinction should be made between raw ASCII text and text which provides some form of markup, i.e. information about the logical and physical structure of the document. In the first case a significant effort needs to be exerted in order to recognise the basic structure of the document and make further elaborations. If the text is marked up, some of this information is already encoded in the document and most of the processes may be easily automated. For a more detailed description of markup languages see section 2.4.3.

• Processing

Once documents are in machine readable form, a number of processing activities need to be carried out in order to produce deliverable objects (Lu 1993). Indexing is one of the most typical processes which is applied to allow a subsequent retrieval, especially from a large corpus of information. Different techniques may be used which range from traditional full-text indexing to more sophisticated techniques based on natural language and neural networks. Today all the indexing and retrieval methods deal mostly with text. The retrieval of other media such as audio, and graphics is mainly based on assigned keywords.

Another issue concerning electronic document production deals with providing an appropriate interface. This typically depends on the kind of the document and on the user tasks. In the previous section, presentation issues related to electronic books have been widely discussed. Hypertext provides another solution for organising and

presenting documents with an intrinsic non-linear structure. Section 2.4.4 will discuss issues related to converting text into hypertext. Concerning electronic documents in general, there are two main presentation strategies: maintaining the page metaphor on the screen (this approach nas been adopted for presenting most of the electronic books, and in desktop publishing systems such as PageMaker); or using the screen metaphor, where information scrolls off from the bottom, as in most wordprocessors. In both cases, a problem which may arise is exchanging these documents. Most of the programs for viewing documents are too specialised for universal use. There is the need of document exchange standards.

At present, a number of packages have been developed as publishing software. Most of them have emerged as an evolution of more traditional desktop publishing systems. Electronic publishing systems supply appropriate tools for creating electronic documents (from scratch or by importing text) and providing suitable layout. They usually incorporate a built-in tool for indexing and retrieval, navigation tools and other user services. Examples of these systems (Wood 1992, Liebing 1993) include FrameMaker, developed by Frame Technology Corporation, with its associated FrameReader, a tool for viewing electronic documents; Ventura Publisher, developed by Ventura Software; Dynatext, developed by Electronic Book Technology; and Folio Corp's Views 3.0.

• Delivery

There are several channels for electronic document delivery, depending on the size of the product, the kind of information, and the storage medium that is employed. The most used delivery strategies include magnetic and optical disc, and networks. CD-ROM is one of the most popular optical storage media; it is increasingly used for distributing large documents databases that do not change rapidly. Some examples of books which are distributed on CD-ROM have been reported in the previous section (e.g. Grolier Encyclopaedia).

Communications networks provide two useful methods of making electronic documents available: remote access, when the material available on a suitable host can be accessed remotely through the network, and downloading, when the material is directly transferred to the delivery environment. Network delivery is not yet widely developed although several initiatives already exist in the field. One reason is that the technology has been inadequate or too expensive; another is the organisational change required to accommodate such a radical change in distribution methods (Tuck 1992a). A number of initiatives in Europe, Japan and the US have emerged concerning the

building of high-speed networks (e.g. NREN, SuperJANET). In addition the integration of broadcasting, telephone and data services on broadband ISDN promises a wider market for electronic document delivery.

At present, a number of network information services have emerged such as Gopher, WAIS, Mosaic which provide transparent access to several databases which are distributed on the network. Significant interest is also emerging for electronic journals publishing (Electronic Publishing Services 1992).

2.4.3 Markup Languages and SGML in the Production of Electronic Documents

An electronic text may be available in several formats: the simplest one is a flat ASCII text. However, there are many programs for writing and manipulating text and each of them stores text in a specific format. In other words, if text is not an ASCII text, then it contains a kind of markup, which can be used for different purposes (writing, printing, exchanging, screen presentation, etc.).

"Markup is the term used to describe codes added to electronically prepared text to define the structure of the text or the format in which it has to appear " (Bryan 1988). Markup varies for different word processors and output devices. The large variety of markup languages causes serious problems when a document has to be exchanged. This is the most commonly experienced incompatibility problem. Standard markup languages provide a way for overcoming this problem, being designed to be application and platform independent (van Raaij 1993).

This section starts with a general introduction to markup languages, focusing on SGML. Then some experiments will be mentioned which show how SGML documents can be processed and translated.

2.4.3.1 Markup Languages

When speaking about markup languages an important distinction has to be made between the logical and the layout or physical structure of a document. The logical structure indicates the document component parts; for instance a document may consist of an introduction, followed by a sequence of chapters and a conclusion. The layout structure indicates how a document looks on the page or on the screen. All the information about the document physical components (e.g. pages), their location in the display surface, the set of fonts used and so on are specified in the layout structure.

While in conventional printed material it is impossible to separate the two structures, in electronic encoded documents the information about the logical structure may be stored independently of how they are represented on the screen (i.e. the layout structure). This is made possible by using *generic* markup languages (van Herwijnen 1990, Bryan 1988), which describe the logical structure of a document by marking the various components or elements. The processing instructions are kept separate from the document and executed by a text formatting program. In other words, generic markup describes a document's structure and its attributes rather than specifying the processing to be performed on it. The same document can be presented in many different ways, according to the stylesheet applied.

There are also other kinds of markup, called *specific* markup languages (van Herwijnen 1990, Bryan 1988), which do not keep the logical and the physical structure separate. Specific markup contains instructions which are concerned with the physical appearance of a document. It is system dependent and requires a knowledge of typography. Typical examples of specific markup are those used by word-processors such as Wordstar, WordPerfect, and by some text formatters such as Script, nroff, TeX.

From the electronic publishing point of view, there are several benefits when documents are available in a generic markup format. These can be summarised as flexibility, speed and efficiency (Smith 1987). Multiple styles and formats may be associated with the same document, many editions can easily be produced, a variety of output forms are possible (CD, microfiche, etc.). All these operations do not require the document to be rekeyed with consequent cost and time savings.

SGML (Standard Generalized Markup Language) is an example of generic markup language. It was introduced in 1986 as an International Standard (ISO 8879). SGML has emerged from the publishing and printing communities, where flexibility of document design is of prime importance. The purpose was to provide a standard way for authors to submit documents in a form that is convenient for professional designers and printers to process (Brown 1989). SGML is appropriate for describing highly structured text. Other components of a document such as diagrams and pictures are managed as external elements that may be arbitrarily encoded. SGML will be further discussed in the next section.

Hytime (Hypermedia/Time-Based Structuring Language - ISO 10744) is an extension of SGML which includes specifications for hypertext, multimedia and time- and space-based documents in terms of their logical structure (Newcomb et al. 1991).

ODA (Open Document Architecture - ISO 8613) is another International Standard for representing and exchanging electronic documents. Unlike SGML, ODA was developed for distributed office systems rather than for publishing. An ODA document is described by a logical structure and a layout structure. They are both hierarchically organised. The content is common to both structures and provides the link between them. ODA specifies codes for describing non-textual objects such as images, tables, etc.

Another interesting group of markup languages are page description languages (PDL), which define how a document has to be laid down on the printed page or on the computer screen. **PostScript** is a page description language, created by Adobe. It defines exactly the physical appearance of a document on the printed page. Postscript files are portable from one system to another; they can be readily transmitted through a communication network to be printed out at the receiving site. PostScript codes are intended to be passed to a laser printer, which translates them into a page image. Adobe has produced an electronic publishing product, named Acrobat (Wood 1992, Liebing 1993), for online viewing of any Postscript document.

While a page description language is suitable for viewing documents on paper or on screen, it is not appropriate during the preparation and production phases of a document because no information about the logical structure is provided; on the other hand, generic markups may be highly useful during these phases. "Maintaining the underlying logical structure of a document can have many advantages for document processing systems. Ideally, the structure reflects the way the author/editor thinks about the document ... the structure must be preserved during the storage and transmission or these advantages are lost" (Brown 1989).

2.4.3.2 SGML (Standard Generalized Markup Language)

Wright (Wright 1992) defined SGML as a programming language designed to write tagging systems. Macleod (Macleod et al. 1991) describes SGML as a system for specifying markup languages rather than a markup language itself, a sort of meta-language. This is made possible by using Document Type Definitions (DTD). A DTD

defines the logical structure for a class of documents. It is a formal definition that indicates which elements occur in the document's content and in what order.

Each element is marked up by a start tag at the beginning and an end tag at the end. The tags specify the *generic identifier.* which identifies the type of the element (paragraph, heading, figure, etc.) and other characteristics, *attributes*, that further qualify the generic identifier. More specifically, the DTD defines:

- generic identifiers of elements that are allowed in a document of a given type;
- for each element, the possible attributes, their range of values and defaults;
- for each element, the structure and its content including which sub-elements can occur and in which order.

A DTD defines a document as a tree, whose root is the document itself. An example of how an element declaration in a DTD could appear is reported below.

<! ELEMENT textbook (front, body, rear) >

Textbook, *front*, *body*, *rear* are all generic identifiers of elements. The element textbook is defined by the group *(front, body, rear)*. Each token in the group is an element which needs to be further defined.

Among the possible attributes of an element, the attribute unique identifier (ID) is particularly interesting. It allows an element to be uniquely identified. This is an important feature of the markup in order to define links.

A number of DTDs exist in the public domain (for instance the British Library SGML starter set, the CALS DTDs, the Electronic Manuscript Standard developed by the Association of American Publishers) and several products, developed for helping users manage SGML documents, are now commercially available. There are editors and taggers (Author/Editor from SoftQuad, Mark-It from Sema Group, etc.), used to add markup tags to unstructured text. There are SGML parsers (e.g. XGML from Exoterica Corporation, etc.), which verify that a document conforms to its DTD. There are also translators from different formats to SGML (e.g. FastTag form Avalanche Development Corporation). A list of products is available in (DeRose 1991) and (Graphic Communication Association 1992).

A number of other standards related to SGML have been defined and others are under development: SDIF (Standard Document Interchange Format), the already cited HyTime, an ISO standard for representing hypermedia information, SMDL (Standard

Music Description Language), DSSSL (Document Style Semantics and Specification Language), etc.

The availability of several SGML products as well as the existence of new standards related to SGML are giving a significant contribution to the development of SGML

Moreover, SGML offers a number of advantages over other markup techniques. One of them is that it is independent of the computer system and of the application. Another important advantage derives from the fact that it is a standard; therefore it facilitates the problem of exchanging electronic documents between different systems. SGML markup is explicit and it can be read without SGML-compatible software. SGML is unambiguous: there are syntactic rules for marking the document in a rigorous way.

Another advantage is that it permits the description of an infinite diversity of document structures as are found in publishing" (CEC, DG XIII-b [nd]). This would not be possible for instance with ODA, because ODA provides a means for exchanging documents that fit a known, necessarily finite range of document architectures and standard content types.

SGML could also be very useful in libraries. "When documents are marked according to SGML, all of the bibliographic elements are identified and tagged ... This means that the basis is already there for database processing and retrieval ... SGML establishes a consistent method for identifying document structure and content, and this may eventually lead to the automation of cataloguing and indexing" (Feeney 1989).

SGML also has its weaknesses. The main drawback is that, in order to cope with a large variety of document types, SGML is quite complex. If a user wants to create a document with a specific structure, he has to know in detail how to design a DTD and how to mark a document according to that DTD, unless an existing DTD is appropriate for such document. Usually the task of document analysis is carried out by a document designer, or someone who is familiar with the documents and their usage (CEC, DG XIII-b [nd]). SGML editors and other similar packages (e.g. SGML translators) are intended to help users in editing and managing SGML documents.

Other weaknesses are associated with the fact that "SGML does not actually specify documents. It specifies DTDs, and incompatible DTDs defeat the purpose of universal document exchange" (Cronk 1993).

SGML applications are more advanced in the United States than in Europe. The largest application of SGML was in the USA with the CALS initiative (van Herwijnen 1990): the US Department of Defence has specified SGML as a documentation standard. Another relevant application was the Electronic Manuscript Project (EMP) by the Association of the American Publishers. The aim of the project was to produce a standardised approach for describing, preparing and interchanging electronic documents. The result was a set of SGML compatible codes.

Among the European projects, there is Formex, an International electronic publishing standard, adopted by the Office for Official Publications of the European Communities, incorporating SGML. Other European initiatives were undertaken by the British Library, which has published the SGML starter set for encouraging the use of SGML (Feeney 1989) and by the European Centre for Nuclear Research, CERN, which have partially adopted SGML as markup language for document preparation.

2.4.3.3 Examples of SGML-Based Systems

In this section different systems will be mentioned whose common aspect is the capability of translating SGML documents into other formats.

One of the most interesting is Dynatext (Electronic Book Technologies 1990), a commercially available electronic book publishing software system. It allows the conversion of existing SGML texts into dynamic electronic books. Dynatext consists of two components: an Indexer and a Browser. The Indexer receives as input a collection of SGML files and associated artwork standard formats, builds a full text index, a dynamic Table of Contents and establishes different kinds of links (cross-references, links to tables, footnotes, figures). The Browser allows access to the electronic book through navigation tools (dynamic Table of Contents and hypertext links). Documents are presented in multiple windows, each displaying a different view (the full text, the active Table of Contents, etc.). It is possible to define style sheets that specify the formatting and visibility of SGML elements and the actions to perform when links are activated. Entity references and attributes can influence the kind of display. The indexing phase takes into account the text elements in the hierarchy. The answer to a query is reported in the dynamic Table of Contents. The indexer supports stop lists and accepts synonym lists. Dynatext runs on SPARC UNIX workstations and can output on any standard X-window device. It exploits existing software: the indexer is based on InterLex, licensed from Brown University; the user interface is

Motif. A new version, Dynatext 2.0, has been developed for Macintosh (Parkinson 1993).

Another interesting system is TOLK (Niblett and van Hoff 1989), a program, written by the Turing Institute, that translates SGML documents into hypertext structures in systems like HyperCard and HyperNews. The program consists of an SGML parser and an interpreted rule language. At first, a document is opened and its DTD is parsed. Then, a set of translation rules (different for each form of output) are used to process it. TOLK is written in C and runs on Sun and Macintosh. The most remarkable limitation is that TOLK does not support the link feature.

A similar system is IDEX (Cooke and Williams 1989), a shell environment which supports a number of processes for managing and retrieving large collections of documents: document conversion, indexing, etc. Source documents encoded in SGML are converted into hypertext format (Guide) after passing through an intermediate format. Documents have a user defined type. The type determines how a document is indexed, how it is displayed on the screen and how it is printed.

These examples, some of them now commercially available systems, demonstrate that it is possible and very useful to exploit information provided by the markup for producing and presenting electronic documents. Hypertext is the representation chosen by some of the experiments mentioned above, i.e. IDEX and TOLK. The difference between them is that TOLK is just a translator while IDEX is an environment which supports a number of other processes for managing large collection of documents. The concept of translating linear text into hypertext will be further described in the next section.

A different approach is chosen by Dynatext; SGML documents are converted to a format which is more similar to a book. A dynamic Table of Contents, cross reference links, full-text indexing are all automatically created during the translation process. Dynatext is the most complete system because it considers a number of issues in detail: the interface design, the user facilities, the indexing process, etc.

2.4.4 The Conversion of Text into Hypertext

Hypertext provides a possible solution for presenting electronic information in a very natural and flexible way. The concept of hypertext is very simple: hypertext consists of nodes of information and links between them (McKnight et al. 1991).

Hypertext is a non-linear way of organising and accessing information. This offers several advantages over a flat and linear organisation of text, namely active links, history mechanisms, the possibility of locating relevant information through navigation tools, etc.

Hypertext is not an isolated world; more and more hypertext functions are now widely used in electronic documents (Quint and Vatton 1992). Therefore hypertext has been considered in this context. A hypertext can be produced from scratch or generated from existing text. In this section, the conversion of existing text into hypertext is primarily considered because it is a specific case of the more general process of electronic publishing described in section 2.4.2. In particular, the processing of the material once it has been acquired, will be discussed because it presents peculiar features due to the particular nature of hypertext.

As McKnight emphasises in his book (McKnight et al. 1991), in order to design a good hypertext system, it is necessary to address the interaction of three aspects of the document usage: users, tasks and information. The best approach for organising a text in hypertext form is to know who are the intended users, what are their needs, and how the system will be accessed by the user. These top-level design decisions have to be made before starting the subsequent stages of development. These design considerations can be generally applied to the design of any electronic document.

This section will describe issues dealing with the acquisition and processing of existing text for producing hypertext. In particular, classes of documents which are appropriate for conversion will be identified, the procedure for converting text into hypertext will be described, and finally some experiments concerning the conversion of linear texts into hypertext will be mentioned.

2.4.4.1 Appropriate Documents for Conversion

Not all types of documents are suited to hypertext representation. Furuta, Plaisant and Shneiderman (Furuta et al. 1989) specify the text organization requirements for achievable and simple conversion to hypertext and they define rules for determining the suitability of text for conversion. The original body of information has to be organised into fragments, the fragments are related to each other and the user needs only a small fraction of them at any time. Encyclopaedias and dictionaries are appropriate for hypertext representation because of their modular structure and interrelations between their components. Manuals, text books and other hierarchical documents are generally well-suited although it could be more difficult to identify the components and their

connections. Novels and other typical linear text are not good for this kind of translation, though interactive fiction relying on non-linear organisations could be considered a new form of novel.

2.4.4.2 The Conversion Process

The conversion problem can be faced in a different way according to how the text is available. If the source text is on paper, a first step consists of producing an electronic version of the document. If the text is available in electronic form, a distinction has to be made between flat or unmarked texts and marked texts. In the first case most of the conversion process has to be done manually; in the second case it is possible to write a program that automatically or semi-automatically translates text into hypertext using information provided by the text markers.

The structure of the text also affects the ability to perform an automatic conversion (Furuta et al. 1989). If documents are regularly structured, an automatic conversion is easily achievable; otherwise a significant amount of manual restructuring is required. This is the case for the conversion of scientific papers into hypertext. For moderately structured documents, a semi-automatic conversion is possible.

The main steps of the conversion process from text into hypertext are:

- fragmentation of the source document into basic units, which will correspond to the nodes in the hypertext;
- definition of the link structure;
- use of some indexing techniques to let the user access appropriate portions of the hypertext document.

In the rest of this section, these steps will be better described.

The raw hypertext defined at this point may be improved by reformatting the text to match the hypertext delivery medium (indenting, centering, widows, and orphans, fonts, etc.). In addition, a number of end-user services may be added to facilitate the use of the systems: browsers, personalisation tools, etc.

Fragmentation

"The process of determining how to modularise a document into nodes is an art" (Conklin 1987). However, the structure of the document may provide useful clues. For instance, for documents which present an evident modular structure such as dictionaries, the choice is quite simple: a dictionary entry is usually selected as the basic unit. For hierarchically structured texts, Glushko (Glushko et al. 1988) suggests

the use of the smallest logical structure uniquely named (i.e. the lowest level subsection). That name represents a good selection key when browsing Table of Contents or as a bookmark for a document location.

Link Definition

Another problem is the definition of links. Sometimes the number of potential links, both explicit and implicit, is enormous. Link structures can be derived partially from the explicit references in the source texts; additional links may be defined which are dependent on the semantics of the text and on the user tasks. The addition of new links to an existing text could generate a hypertext version of information which violates the original author's intentions (Nielsen 1990). "The challenge for hypertext designers is to understand the user's task and to support links that follow from some model of the user's need for information in some particular context" (Glushko 1989). Many studies have been done on the problem of automatic link definition. Bernstein (Bernstein 1990) suggests a solution: a simple "apprentice" that proposes a number of "promising" links to the hypertext author. The apprentice is fast and links proposed are often useful.

Indexing

The problem of Information Retrieval (IR) in hypertext has been widely discussed in the literature (Halasz 1988, Frisse 1988, Frisse 1988a, Frisse and Cousins 1989, Brown 1988, Lucarella 1990, Frei and Stieger 1992, etc.). Navigation, which is the main access technique in hypertext, is sufficient only for some classes of applications, e.g. online interactive presentation, display oriented representation tasks (Halasz 1988). A query-based access mechanism is required when the information space is large, unfamiliar and heterogeneously structured.

As Lucarella points out (Lucarella 1990), hypertext and conventional text retrieval serve different goals and it is reasonable to combine them. Many approaches are possible in order to achieve such integration.

Frisse (Frisse 1988, Frisse 1988a) proposes a method for combining keyword search and hypertext navigation in his application (the conversion of "part of a manuscript of a medical textbook into an experimental hypertext handbook using Xerox's NoteCards"). The query mechanism allows the user to identify the best set of cards for successive browsing. The indexing process takes into account the information inherent in the linking structure to perform a more semantically meaningful search. The basic principle is that if two nodes are linked, then it is probable that their contents are related in same way.

Lucarella (Lucarella 1990) presents a model for a hypertext-based retrieval system using plausible reasoning (the retrieval task is treated as an inference process relying on uncertain knowledge) and proposes an approach for the integration of search facilities in hypertext systems.

2.4.4.3 Existing Conversion Experiments

This section describes two groups of systems: systems which provide tools for converting text into hypertext and systems which are the result of such a process. The Super-Library and SuperBook systems belongs to the first group.

The *Super-Library system* is a prototype developed within the more general Super-Library project mentioned in chapter 1, which represents the context for this research. The Super-Library system supplies an environment to support the conversion of text into hypertext (Catenazzi and Argentesi 1991). It consists of two components: an Authoring Environment and a Browsing-Retrieval Environment. The Authoring Environment provides a number of tools for transforming linear marked text into hypertext; it also allows several related hypertext documents to be integrated. The conversion process follows the steps previously described. First of all, the logical units are identified and some markers are inserted into the source ASCII text. A parsing program takes as input the marked text and builds the hierarchical hypertext, consisting of nodes and hierarchical links. Transversal links are added using a specifically developed tool. The resulting hypertext is then full-text indexed. This conversion process is applied to one document at a time. At the end, the documents are integrated. Common indexes are created to enable a full-text search on the whole collection.

Once the conversion process is complete, the document collection is made accessible in the Browsing-Retrieval Environment through navigation tools and classical Boolean search techniques. The document collection is graphically presented as a shelf of active documents. When a document is opened, its content is shown in a three window configuration. The interface design takes inspiration from the SuperBook project. The three windows respectively contain the table of Contents, the Text, and the search mask.

The Table of Contents window has three purposes: firstly, to show the contents of hierarchically structured documents; secondly, to provide hierarchical links to the document contents; thirdly, to show the results a of search: the frequency of occurrences of search terms is indicated next to each entry. The Text window is a

scrolling window which contains the current selected logical unit of text; active links are embedded in the text. The Search Window is used to define Boolean search queries.

SuperBook provides an environment for converting linear text into hypertext (Remde et al. 1987, Egan et al. 1989, Egan et al. 1989a, Egan et al. 1991, Landauer et al. 1993). The systems goals were: "to use computer-based enhancements to improve the usability of conventional documents" and "to implement the enhancements automatically with the minimal human intervention". SuperBook consists of a preprocessor and a browser. The preprocessor takes as input ordinary texts in a standard markup language (Troff, Scribe, Interleaf) or in flat ASCII with heading markers, analyses them, and automatically builds a formatted text data structure, a full-text index and a dynamic Table of Contents. When the preprocessing is complete, users can browse documents interactively. The information is presented in a multi-windowed screen with powerful search, navigation, annotation and display facilities.

A number of hypertext systems have originated from the conversion of text into hypertext. Some of them are briefly described below.

One of these is the creation of a compact-disc version of the *Engineering Data Compendium* (Glushko et al. 1988, Glushko 1989), a multi-volume scientific and engineering encyclopaedia containing information about human perception and performance. The Engineering Data Compendium is a very good candidate for a hypertext representation because of its structure. It essentially consists of small entries which deal with well-defined topics. The entries have a regular internal structure. They are uniquely named and numbered. They contain explicit and implicit cross-reference links. Figures and tables are also included. The user interface supports different methods for finding relevant entries. These methods include browsing Tables of Contents, Index and other hierarchical access points, and searching (full-text Boolean search). The browser provides different facilities: it marks the exact position in the hierarchical structure, allows movements in the hierarchy, expands and closes items, indicates whether an item can be (or is) expanded or not, and marks the last item expanded.

Another work, quite similar to Glushko's one, is reported in (Raymond et al. 1987, Nielsen 1990a) and deals with the hypertext representation of the *Oxford English dictionary* (OED). The OED consists of 16 books (12 of the original version and a four

volume supplement published more recently); the basic unit is the entry, variable in size and structure. Hypertext is a good access mechanism for this application because of the modular structure and the numerous cross references within the dictionary.

Another interesting example of document conversion is *Hypertext on Hypertext* (Furuta et al. 1989). It is the result of the conversion into hypertext of eight scientific papers, published on July 1988 Communications of the ACM (CACM), drawn from the Hypertext Conference held in North Carolina in 1987. Three separate hypertext versions were prepared for Hyperties, HyperCard and KMS. The conversion process required a consistent manual intervention for rearranging texts, with minimum cross referencing across papers; new links were created; overviews were added. Such experiments demonstrate that, even though some parts of the conversion could be automated, the bulk of the process needs a significant manual effort.

2.5 The Electronic Library

In section 2.3, the concept of electronic books has been widely discussed and presented as a way of overcoming most of the limitations of paper books. Because of the increasing development of electronic books, there is a consequent demand for mechanisms for organising and accessing them. Libraries need to adapt themselves to cope also with these new entities, electronic books.

Over the last decades, technological development had a deep impact on libraries. Nowadays many libraries use electronic support for operations such as acquiring and cataloguing material, searching and retrieval. Information technology is an aid for both the librarian in order to organise the material and for the user in order to gain access to the broad storehouse. Information is still physically stored in the library. This represents an intermediary step in the process of library automation which leads to a completely electronic library, where a timely provision of selected full-text materials to individuals when and where they need them is guaranteed.

This section illustrates the main features of an electronic library, through the evolution of the paper library towards the full-text electronic library. Some projects dealing with full-text electronic libraries will be mentioned.

2.5.1 From the Traditional to the Automated Library

In the past, libraries were places where documents were preserved and collected (McGarry 1991). Access to the material was restricted to a small group of people. The concept of the library as a service for a community of users emerged later. Two factors were responsible: the invention of printing movable type, which allowed multiple copies of a document to be produced, and the fact that more people were literate. The library as it is today is the result of a literacy process which causes an increasing need for a service which provides information.

Traditional library services are divided into two areas: technical services and public services. The first category includes operations such as acquisitions, cataloguing, shelving and binding. These are backroom operations; patrons are only aware of the final result. The public services include circulation, loans, reference services and user assistance.

Library automation has been a gradual process which began in the mid-1960s (Hildreth 1985) when computers were first used to produce catalogue cards or book purchase orders. The next step was the introduction of machine readable catalogues and then cooperative catalogues, which were the result of agreements between a number of libraries to share their work. One of the most important developments was the definition of the MARC (MAchine-Readable Cataloguing) standard for exchanging machine readable catalogue records (Arms 1990a). This was developed by the Library of Congress in the late 1960s and later ratified and adopted as a standard in early 1973 by ISO. In the third phase, computers were applied in support of a number of tedious and labour-intensive operations of the library: acquisition, serials control, circulation, etc. Commercial systems for searching reference databases online also emerged (e.g. DIALOG) although, because of the training required, usage was initially restricted to librarians. Up to that time, library automation can be seen to have been a key issue with regard to internal, back-office services but had not affected user interaction with the library itself.

Online public access catalogues were the first major development to bring the benefits of automation directly to the user. An Online Public Access Catalogue (OPAC) is a library catalogue accessed via a computer terminal for the benefit of library users (Walker 1989). The current generation of online catalogues represents an improvement over manual catalogues. They allow the user to know the real availability of documents and they offer major access points. Catalogues gradually moved from first generation systems (second half of 1970s), which offered poor facilities such as records only consisting of author, title and location, to advanced systems, second generation catalogues, which provide keyword facilities. The tendency is toward more flexible, adaptive and responsive systems (third generation) which offer browsing and navigation facilities, ranked output subject searching, etc. An interesting example of a third generation catalogue is the Book House system (Pejtersen 1992), an interactive, multimedia, online public access catalogue designed to support casual, novice users in information retrieval.

In addition to online systems for accessing bibliographical information, CD-ROM technology is gaining importance. CD-ROM provides a good medium for the storage and retrieval of data that does not change quickly. CD-ROM and online services cover different needs and they can be considered to be in some way complementary (Vogt 1992). Online services can provide more current data, but they always need a network. On the other hand, CD-ROM cannot be updated in real time and the storage capacity is

limited; but CD-ROM workstations are portable and robust; therefore, they can be used even in places where telecommunications are unreliable or non- existent.

Library automation is having consequences also on the librarian's role (Callaham 1991, Lyon 1990). The librarian is becoming the agent responsible for bridging the gap between the new technological innovations in information storage and retrieval and the library user. The traditional role of providing organisation and access to information is now becoming a marginal role. Most of the routine tasks are now executed automatically. The librarian's main role is to educate people to efficiently and effectively use these new technological innovations, thus overcoming resistance to change. People are afraid of the computer itself (its jargon) and are worried of damaging it or making mistakes. The librarian has to understand how the technology works in order to help users formulate search queries in the best way and assimilate information which has been retrieved.

Although library automation has brought enormous advantages over the traditional paper library, namely greater efficiency in the housekeeping processing and the possibility for a user to quickly search information using OPACs, it also has a number of disadvantages (Wu et al. 1993):

- OPACs contain only bibliographical information which could be insufficient for deciding whether the retrieved material is relevant;
- users need to go physically to where the printed material is located to get it;
- usually OPACs do not contain information about journals; there are separate sources (indexes) for retrieving journals;
- the printed material occupies considerable space.

Most of these limitations have been already overcome in some automated libraries. For instance, if the library collection is remotely located, some strategies already exist for document delivery. The traditional and most widely used method is based on postal mail; fax technology represents an interesting alternative for speeding up the delivery process: images of the requested material are transmitted from a document supply centre (the library) to the user. Other methods for document delivery exist (e.g. e-mail, ISDN network), but they require the paper document to be scanned and converted into digital form. These methods could be better applied in a full-text electronic library, as described in the next section.

2.5.2 Towards the Electronic Library

The electronic library can be thought of as a new strategy for delivering information where the full-text of documents is available online. The library can be accessed remotely from any location through a network using personal or portable computers. Universal accessibility represents the most revolutionary aspect of the electronic library. The contents of the library, virtual books, no longer suffer from the constrains of their physical counterparts and can be replicated as many times as required. The library itself has infinite space and there is no limit to the number of books the library can contain. Individual libraries may be interconnected in a transparent way.

When organising electronic libraries, most of the administrative and technical operations performed in traditional libraries have to be reconsidered:

- many of the operations concerning the *collection management* that are normally performed in physical libraries, e.g. binding and preservation, shelf maintenance, do not exist in electronic libraries (Rawlins 1993), although there may be new administrative and technical functions that have to be carried out, such as system maintenance;
- other issues deal with the *acquisition* of material. Material can exist in different formats and conversion tools are usually essential parts of the acquisition process in electronic libraries. Activities such as ordering of material may also be carried out directly in electronic form (EDI ordering);
- once the material has been acquired in electronic form, a number of processing activities are carried out in order to store it in the library database: operations such as *cataloguing* and *indexing* may be significantly facilitated by the fact that full-text of material is available in electronic form. A number of artificial intelligence techniques and language techniques have been exploited in fields such as cataloguing, indexing and abstracting (Morris 1992, Alberico and Micco 1990); operations which are traditionally carried out manually may be automatically or semi-automatically executed in electronic libraries. SIMPR (Structured Information Management: Processing and Retrieval) (Gibb and Smart 1991, Gibb 1993) is an example of a knowledge based system aimed at indexing and retrieval from large volumes of information. CLARIT (Evans et al. 1991) is another system which exploits artificial intelligence techniques for indexing;
- other issues deal with *public services*: how to provide copies of documents; how to display them; how to inform users; how to assist users; how to manage search and retrieval, taking into account gateways to other library systems; how to protect

copyright; and how to manage loans, which can be interpreted as either temporary provision of copies or free copy delivery according to the system strategy. Assuming that users can access both local and remote document supply centres from their workstation, they need to know how to choose an appropriate service, how to choose the right database, how to select a telecommunication network, which are suitable application packages to manipulate information, how to process the information delivered, etc. Intelligent interfaces are needed in order to cope with such variety of problems. The computer gradually replaces the librarian's role as intermediary between users and the library. Expert systems are being developed in order to simulate the human reference librarian in electronic libraries (Morris 1992, Alberico and Micco 1990): they interact with users in order to understand the user's query, they help in understanding the result of a search, they provide general information about the system, etc. Plexus (Brooks 1985, Vickery at al. 1987) is an example of an expert system which provides an intelligent interface, allowing the end user direct access to information sources, without recourse to the mediation of a reference librarian.

The electronic library could provide facilities and benefits which were almost impossible in a traditional library environment. For example, a mechanism for keeping people abreast of new and modified information could be an useful tool offered by the electronic library (Yankelovich 1990). The user profile is an important aspect that an electronic library could consider (Goodram and May 1991). Knowledge about user needs may be used in order to optimise the retrieval strategies and the acquisition policy. The concept of user profiling has been incorporated for instance in the RightPages™ Service (Hoffman et al. 1993), an electronic library which provides an alerting service to inform users about new publications in their field of interest.

Figure 2.1 summarises the main activities involved in electronic libraries compared with those of a traditional library. Many tasks are similar, although they are performed in different ways as a consequence of the characteristics of the operational environment.

A number of issues have to be addressed in order to ensure that the idea of a full-text electronic library can be realised (Arms 1990). First of all, there is an increased demand for online and archival storage. The decreasing cost of memory and the development of data compression techniques have made it possible to store economically gigabytes of data.

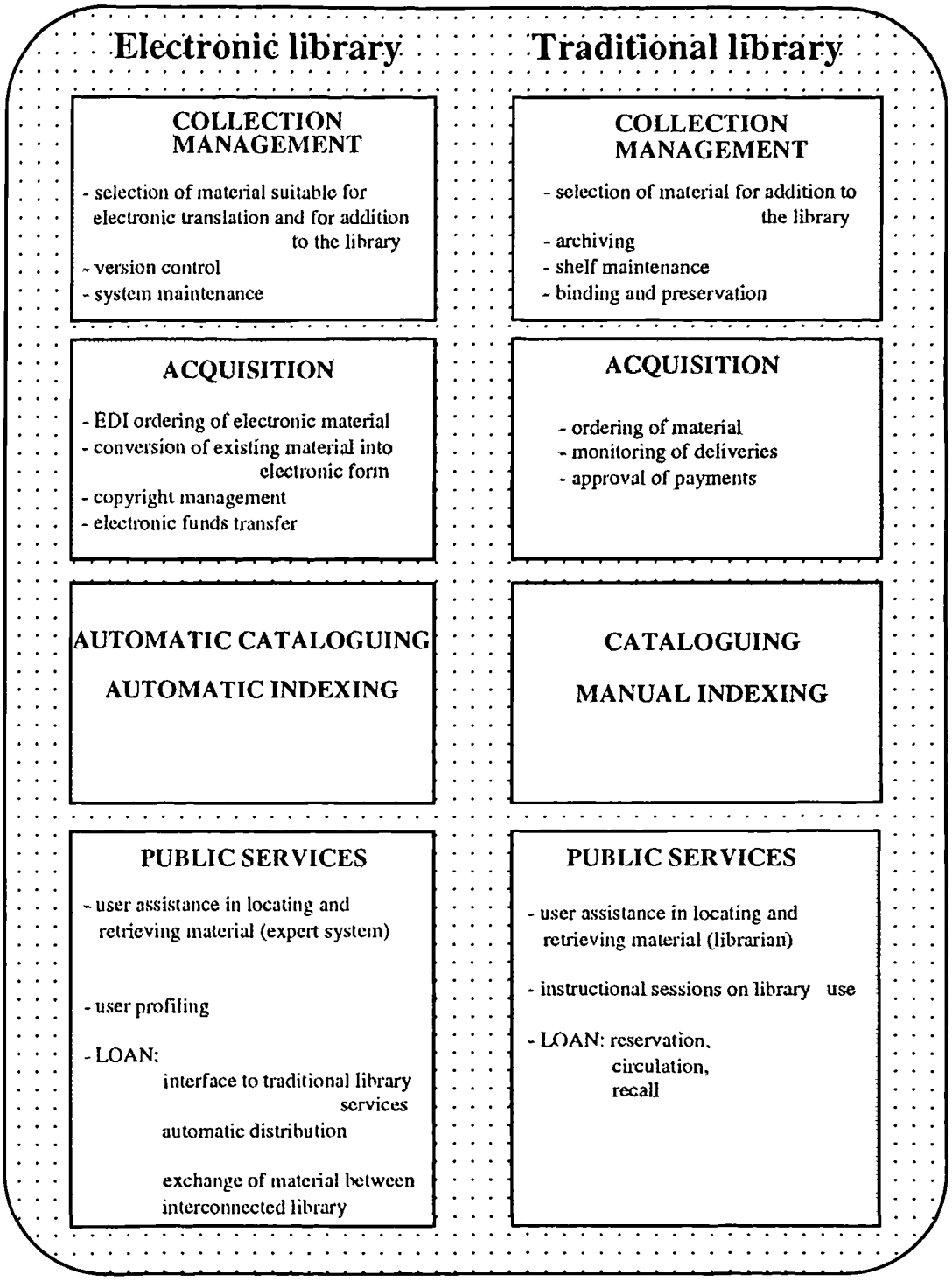


Figure 2.1: A comparison between a traditional and an electronic library

At the same time, there is the need for more powerful personal computers linked to higher speed networks for managing a number of public services: bibliographic search, document ordering and delivery (Tuck 1992, Tuck 1992a). Current technologies include fax, electronic mail, CD-ROM and file transfer. Database search exploits either

the interactive online or CD-ROM technology. Fax has been currently adopted as a means for document delivery by many systems such as ADONIS, ELNET, and is the cheapest way to transmit images. On the other hand, fax transmission is slow, the quality is not always satisfactory and does not provide mechanisms for linking it to a document request facility, which is instead made possible by using electronic mail.

Another factor to be considered is the role of standards for:

- representing the structure of documents: e.g. ODA, SGML which are designed to facilitate the exchange of complete documents;
- cataloguing records: MARC is widely used for the exchange of bibliographic information although this has been generally restricted to monographs; and
- information retrieval: a standard protocol was approved by NISO (National Information Standards Organization) and ANSI and is now American National Standard Z39.50/1988; the same has been proposed as an ISO standard.

2.5.2.1 Current Projects

The purpose of this section is not to present a complete overview of the current projects in the area of the electronic library, it is just to give a general idea of the current line of research .

There are several experiments and projects which are related to the concept of the electronic library. Some of them are large scale projects which originated from the need of existing libraries to organise and distribute material in electronic form. These systems are the result of an evolutionary process within traditional libraries in academic environments and fully implement the idea of the electronic library presented in the previous section.

There are a number of other works which are still connected to the idea of the electronic library, but consider only some aspects. Some experiments deal with hybrid systems, i.e. systems which are not completely electronically based, but where the paper version is still an important part. Another group of projects are information services distributed on the network. Finally there are small scale projects which consider mainly interface design issues for the documents presented to the users.

Full-Text Electronic Libraries

The CORE project is a large collaborative project (Lesk 1991, Egan et al. 1991a) which involves the participation of many organisations: the Mann Library, the American Chemical Society (ACS), BellCore and the Online Computer Library Centre

(OCLC). The goal of CORE (Chemistry Online Retrieval Experiment) is to create a large database of journal literature (ACS journals) and reference works in the area of chemical sciences to make these available to the Cornell Campus University. The project aims to develop a prototype system for the storage, searching and displaying of primary scientific journal data in electronic form together with their graphical information. Specific consideration is given to the acquisition phase of the material and to the evaluation problem. Data arrives from two sources: microfilms and magnetic tapes. Microfilms contain copies of the paper journals and the magnetic tapes contain the primary journal text, indexes and abstracts. Microfilm material is scanned and processed in order to get images of the original paper and to isolate graphics and figures to be displayed within the text. ASCII texts are well tagged and the processing does not require a big effort; the text is then indexed. The material is accessed using different interfaces and retrieval systems: an ASCII-based interface, SuperBook (Egan et al. 1989), which is based on full-text search and an image display system, Pixlook, which supports a full Boolean fielded search capability. The evaluation considers two different situations: performance in normal use and performance in controlled tasks. In the first case, user behaviour is observed and recorded while he is doing his own task. In the latter case, users are given specific tasks and their performance is evaluated. A comparison is made between the different interface systems: ASCII display, page image display and paper version.

The same ACS journals have been used as the basis for another similar project, called CODA (Kirstein 1993).

Mercury is a research project (Arms and Michalak 1990), developed at Carnegie Mellon University. The project aims to demonstrate that current technology (high speed networks, high resolution screens, multimedia facilities) and the techniques for processing electronic documents, make it possible to build full-text electronic libraries. Mercury is also evaluating a number of different approaches for acquiring and storing documents, e.g. scanning documents and saving them as images or capturing documents in machine readable format.

Elinor (Electronic Library - Information Online Retrieval) is a research project under development at the Information Centre of De Montfort University Milton Keynes (Wu et al. 1993, Arnold et al. 1993). The project objective is to create a large electronic text and image database of books, journals and course material, which can be directly accessed by students and staff. Particular attention is given to issues related to the

conversion of printed material into electronic form, to the consequent storage requirements and networking problems in a multi-user environment.

The NCSU Digitized Document Transmission Project (Casorso 1992) is a collaborative project between North Carolina State University Libraries, the National Agricultural Library and several land grant university libraries, which aims to explore the technical, procedural and administrative issues involved in developing network-based document delivery systems for library materials.

Hybrid Information Delivery Systems

There are a number of systems which could be considered as hybrid systems as they are not completely electronically based. They mostly deal with journals publishing.

ADONIS (Articles Delivery over Network Information System) (Electronic Publishing Services 1992) for instance, is a system devised by a consortium of scientific publishers which involves nearly 500 journals. When a new journal is published, details of each article are recorded in the index, the journal pages are scanned and saved on CD-ROM. Documents could be searched over the network using online indexes. Document delivery is mainly based on group four fax machines, offering the user with an on-demand service. A further evolution of this system will lead to the integration of the document databases directly to the network so that documents could be automatically transmitted on demand to the requested user.

Similar initiatives are:

ELNET (King 1992), a database in Japanese, composed of 37 newspapers and more than 143 magazines all digitised on optical support;

Tulip, a cooperative project testing systems for networked delivery and use of journals (Electronic Publishing Services 1992);

McGraw-Hill's Primis, a printing on-demand system, which assembles text-books matched to a requested format (Electronic Publishing Services 1992).

Network Information Services

Several network information services have recently emerged; they provide transparent access to different databases accessible through the Internet network (Singh and Meadows 1993). Examples of these systems include Gopher, WAIS (Wide Area Information Server) and WWW (World-Wide Web).

Another interesting project is Gutenberg. The main purpose of the project is to encourage the creation and distribution of electronic texts. Specifically, the goal is to provide a collection of the most extensively used books by the year 2001 (Hart 1992). The underlying philosophy is to make texts available in a form which is simple and easy to use and which can be delivered across a wide range of platforms in order to support an extremely large audience. The format chosen for the electronic texts is called "Plain Vanilla ASCII", which is the low set of the American Standard Code for Information Interchange. This is quite attractive as 99% of current hardware and software already has the functionality to read and search such files.

One of the most ambitious proposal is Ted Nelson's Xanadu system (Samuelson and Glushko 1991). His aim is to design a network-based hypertext system capable of storing and providing access to the world's complete stock of textual material (the Docuverse). Each piece of text is uniquely referred and is only held once in the system irrespective of the number of documents in which it may occur. Copyright issues arising from the use and distribution of texts are deeply considered and an intellectual property scheme is proposed.

Interface Design Issues

Interface design is a very important issue in the context of an electronic library. This mainly concerns the presentation of electronic documents and the simulation of the library environment. An interesting approach is to present an electronic library as a virtual library (Saunders 1993), i.e. a library where services and resources are represented through a visual and spatial interface. The concept of a virtual library is based on the more general concept of virtual reality. The user is immersed in an electronic environment which simulates reality by allowing interactions such as movement and tactile control. In a virtual library a user can interact with collections of documents as s/he would in a physical library.

Several prototypes have been developed which focus mainly on the interface design issues. Two examples are presented below.

The WALT project (Frisse et al 1991) is a prototype interface built on top of a hypertext system. It is composed of several elements which permit the user to: examine the document; navigate it; retrieve information and use relevance feedback. The originality of its approach is the design of navigational tools following the shelf

metaphor. Two components of this interface are particularly relevant, the Book Shelf and the Spine Book. The first one offers a way to examine a collection of books through a graphical representation as in a physical library. Book Spine provides the reader with an additional clue while reading the book by giving a graphical representation of the number of book chapters and their relative size.

Another example which is worth mentioning is the "Active Library on Corrosion" (Bogaerts and Agema 1992), a large CD-ROM-based hypermedia system on corrosion engineering, produced by MIPS research group together with Elsevier Science Publishers B.V. and now sold in Europe and United States. It consists of a collection of corrosion case histories, a dictionary, several databases with specific material information, a number of different tools for analysing frequently occurring corrosion problems and four books which are standard works in the field. The system is quite specific but presents interesting tools which have a more general applicability (navigation, backtracking, searching, printing capabilities). Books are made accessible through the provision of a Table of Contents and a Subject Index. A number of typed links are provided in order to expand the document to reveal additional details in the text, to access cross-referenced information and to display additional information in pop-up windows.

Another system which focuses on user interface design for electronic libraries is Project Envision (Fox et al. 1993). This project aims to build a user-centered database from the computer science literature, initially using the publication of the ACM. The system interface design is based on the results of interviews of potential users, i.e. experts in library, information and computer science, in order to understand their needs, to know their perception of existing information systems, and to collect recommendations. In addition, a formative usability evaluation of the interface was conducted before developing a prototype.

This research project is also interesting because it has led to the definition of a set of nine principles for digital libraries, covering issues of representation, architecture and interfacing.

2.6 Conclusions

This chapter reports the result of a literature review in several connected areas. Two topics have been mostly discussed: the concept of the book and the issues connected to book publishing.

The book has been firstly considered in its traditional paper form, in order to extract features and properties which are dependent on its intrinsic nature (mainly structural and layout features). Moreover, books have been considered in the electronic environment, which allows a number of new features to be added, mainly flexibility and interactivity. Hypertext links are a typical facility which may be added in order to increase flexibility and provide different navigation paths. A standard way for presenting electronic books does not exist. Many different approaches have been adopted which depend on factors such as the nature of the book, the intended users and their needs. Most of them are based on the idea of following the book metaphor, as it provides a familiar and well-known model.

The second topic which has been focused in this literature review is the book publishing process. This process is strongly affected by the delivery medium (paper or electronic). Whereas paper publishing is governed by well-established patterns, electronic publishing is a relatively new area and a number of issues emerged. The major issues related to electronic publishing involve the markup format in which the material is originally available and the interface design. The use of standards is an important factor which has been addressed in order to maximise the possibility of exchanging and disseminating documents. The conversion of text into hypertext has been discussed in this context as it is a valuable example of how an electronic document may be produced. Of particular interest are some examples of systems for converting machine readable text into hypertext, e.g. Super-Library which represents the basis for this research.

The concept of the electronic library has been discussed as it provides an important application of the potential of electronic books. Whilst the use of computers in libraries is becoming common practice and a number of operations are now performed automatically, the full-text electronic library is still a young concept. However, several experiments demonstrate that the time is ripe to make this idea a real and concrete object. The most evident obstacles are not technological factors; they are rather the need to modify existing organisations and procedures, developed over centuries of experience with storing and disseminating information in printed form.

The main implications of the literature review on this research will be discussed in the next chapter.

Chapter 3:
Rationale for this Research

This chapter represents the bridge between the review of the main concepts considered to be relevant to this research (see chapter 2) and the design of a specific model of an electronic book, described in the following chapter. First of all, the general context (the publishing process) is presented, extending the synthetic perspective proposed in chapter 1; the focus of research is then clarified inside this context. Finally, how the analysis of the literature affects design is described .

3.1 The Publishing Process

In the previous chapter, the publishing process has been discussed both from the paper and electronic point of view. A general overview of the traditional paper publishing process is presented in figure 3.1. The main actors who are involved in this process are:

- the author(s), who actually writes the document;
- the publisher, who is responsible for the selection of the material which has to be published and for the editing activities which transform the original manuscript into the resulting paper publication;
- the library, which acts as a repository where paper publications can be borrowed or consulted;

- subscription agents, who manage the subscriptions of journals to libraries and other institutions;
- the book shop, a place where paper books may be viewed and sold;
- users, who can access the published material both from the library and the book shop;

Editors, designers, and typographers, who are also involved in the publishing process, have not been mentioned explicitly. It is assumed that the publisher incorporates these other roles.

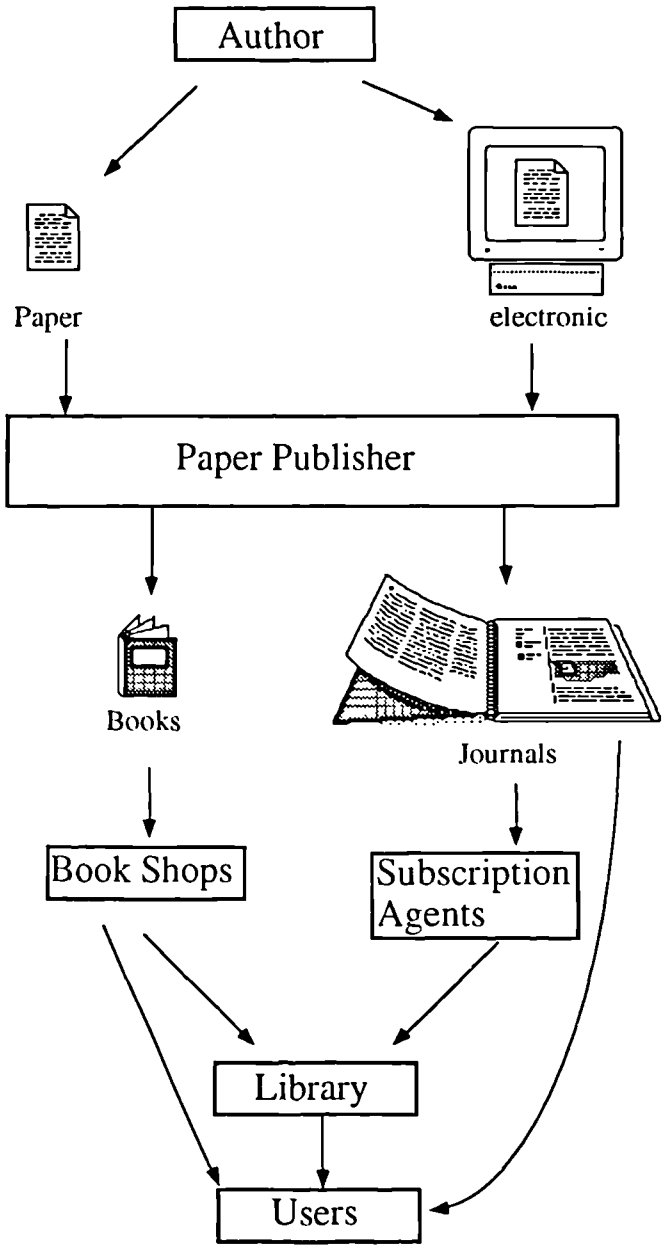


Figure 3.1: An overview of the traditional publishing process

The author can submit his manuscript to the publisher in paper or electronic form, as described below. The important fact to note is that the result of the publishing process of both journals and books is still in paper form.

Other actors, which operate as a part of the distribution channel, are the book shops and subscription agents. The book shops sell books both to readers and to libraries. The subscription agents manage the subscriptions that libraries take to journals. Their roles are important because they provide economies of scale based on one to many rather than many to many relationships, thus simplifying the task of acquisition for libraries. Libraries do not need to contact multiple publishers in order to acquire their publications; but they directly interact with subscription agents and book shops who are in charge of collecting material from different sources.

Users can borrow and consult the library material, buy books at the book shop or directly receive journals to which they have subscribed. The physical nature of publications represents a constraint for getting them. In fact, in order to be able to read or consult books or journals, users need to go physically to the library or the book shop or they can receive them by mail.

Libraries provide physical and logical organisation of materials and tools for facilitating access to these materials. If we consider the traditional public library, we can identify a number of advantageous features:

- A library is a repository for organised collections of material shared by and accessible to a large community of people.

- It provides a number of tools for accessing and locating material in a standard form (catalogues, indexes, librarians, etc.).

- It is a service managed on behalf of a community, thereby freeing individuals from the need to acquire and organise relevant literature.

- The library is a cultural resource which can respond to different user needs, both currently and in the future.

- The library can demonstrate economies of scale such that costs are shared across the community of users.

The traditional view of publishing is however being changed by information technology. More and more information is available in electronic form and several distinct classes of tools already exist for handling it, as described in the previous chapter. Nowadays, the publishing process involves both paper and electronic publications. A general perspective of the current situation is synthesised in figure 3.2, representing the main actors who are involved in the publishing process and the main relationships that may exist between them.

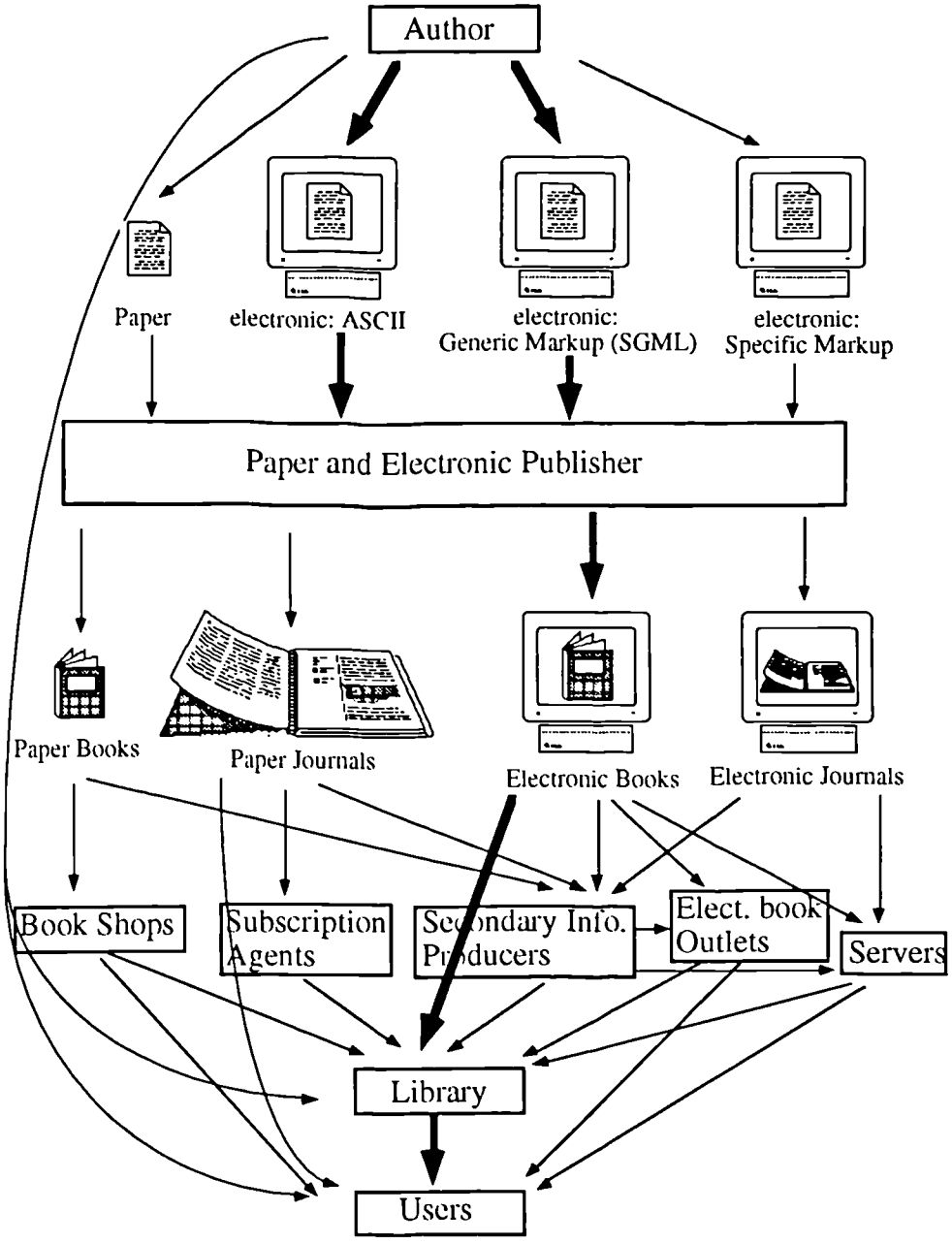


Figure 3.2: An overview of the paper and electronic publishing process

Most of the actors are the same as described above; the difference is that they now have to manage both electronic and paper texts. There are publishers who still publish in paper form; others who publish in electronic form. The new actors which emerge in this context are Secondary Information Producers, Electronic Book Outlets, and Servers, as described below.

Following the scheme of figure 3.2, the author is taken as the starting point. It is assumed that the principal mechanism for authors to disseminate the results of their work will continue to be through publishing houses (although the nature of these publishing houses may change). One less likely possibility is a direct connection between the author and the library and/or the user thus bypassing the publisher and placing the responsibility for publishing on the library or on the author. Unless the library incorporates the roles of a publishing house, the significant consequence of this strategy would be that any document could be published without quality control. This is a highly topical issue at present, but one which falls outside the boundaries of this research.

It is possible to identify different modes of publication through publishing houses as it is unlikely that all publishers would adopt the same approach. Figure 3.2 presents several options concerning input and output formats. Firstly the possible input formats are considered. There may be publishers who continue to accept paper manuscripts; publishers who allow their authors to submit their manuscript in simple ASCII form; others who adopt specific markup languages, provided by the majority of word processors; others who may adopt a generic markup language, such as SGML. This final option has the greatest potential as it avoids the problems caused by incompatible systems and markup languages and therefore saves production costs, as well as providing greater flexibility in presenting and exchanging information (see section 2.4.3). "The development of SGML came about largely as a result of the desire of publishers to have a standard means of preparing electronic documents for publications" (CEC, DG XIII-b [nd]). The use of specific markup makes it difficult to exchange documents with different systems; on the other hand, the use of ASCII solves the problem of exchanging files between incompatible systems, but it is a limited solution as it does not provide any information about the document structure and layout.

The result (output) of the publishing process can be on paper or electronic support. Some publishers will continue to publish in paper form. In this case the material may

be distributed and used directly in this form, through the traditional distribution channels (book shops and subscription agents), or may need to be converted from a paper version into an electronic one, for example, using scanning techniques.

Two new actors have emerged which are responsible for managing electronic publications: the Secondary Information Producers, Electronic Book Outlets, and Servers.

The Secondary Information Producers emerge mainly from the need to have a new intermediary between the Publishing Houses and the Library, for certain publications which will be made available in electronic form. This category covers a wide range of information processing activities including indexing, abstracting, cataloguing and converting from one form into another. Some of these functions, such as indexing and abstracting services, already existed in the traditional publishing environment. Of particular interest to this research are services related to conversion of electronic and paper book formats. This may be required because of the different publishing strategies and technologies adopted by the Publishing Houses. Not all publishers may have appropriate technology in order to produce books in an electronic format acceptable to a library. The role of the Secondary Information Producers is therefore to provide a new type of service which is responsible for converting the material acquired from individual publishers into a format suitable for libraries, Electronic Book Outlets, andServers. This may include conversions from paper sources if the material is only available on paper, or adaptation of the electronic format if the original format is not the required one.

Electronic Book Outlets are book shops of electronic books. These books may be published directly in electronic form, or may be produced by Secondary Information Producers, as said above.

Another new actor in the context of electronic publishing is represented by servers. They include all the online services which allow users to access to electronic information of different nature from their own workstation. They may include for instance Bulletin Boards, Lists, News, Electronic Journals, and a number of distributed information delivery systems such as WAIS and Gopher. Servers, Electronic Book Outlets, and electronic libraries (considered below), are the main ways for users to access electronic information.

Libraries are becoming more electronically based. During the last 20-30 years there has been steady progress in the automation of operations which were previously managed in a manual fashion. Computers were first applied to internal, back-office services in support of a number of tedious and labour-intensive operations of the library: acquisition, serials control, circulation, searching reference databases, etc. Online public access catalogues were the first major development to bring the benefits of automation directly to the user.

As we move towards the prospect of a predominantly electronic library we should extend some functionalities while maintaining many features of the physical library. In an electronic library the full text of documents will be available in electronic form (image or character based). In addition, the library will be accessible remotely from any location through a network thus extending its utility to users. An electronic library may also be based on a number of individual libraries interconnected in a transparent way (a meta-library).

When organising electronic libraries, most of the technical operations performed in traditional libraries have to be reconsidered. There are issues related to the acquisition and processing of the material, to cataloguing and classification, and to public services which include how to provide copies of documents; how to display them; how to inform users; how to assist users; how to manage search and retrieval, etc. There are therefore major implications for the acquisition, processing and presentation of electronic books.

Books are the traditional way of delivering information. It is reasonable to assume that current users of electronic books will have already read paper books; therefore they are already equipped with a mental model which can be mapped directly onto a computerised version. This implies that they should be able to use at least the basic features with the minimum training and effort (Burrill and Ogden 1989).

In particular, literate readers, who are the target audience of this research, know how to read books, how to use a Table of Contents, how to use an Index, etc. By maintaining the same model on screen, access to electronic information can be facilitated. Instead of formulating complex queries in order to have information on the content of the book, a representation of the book itself is available which can be consulted like a real book. This approach helps to overcome some of the limitations inherent in electronic support, such as the loss of the physical aspects of the book, e.g. its thickness, which can be used as an orientation clue. Nevertheless, there is another important group of difficulties which are connected to technological shortcomings,

mainly the limited screen size and resolution, that make reading an electronic book less pleasant than a paper book.

This section concludes by considering possible consequences of the rapid development of electronic publishing.

The most remarkable are the possible implications on the copyright. Copyright gives creators and publishers the tools to protect their intellectual property and their financial interests. Electronic publishing could require some adjustments in the copyright law, as it provides a number of new capabilities not available with the paper support (ease of replication and transmission, ease of modification, etc.).

Other consequences of the electronic publishing development are changes in the structure of the industry caused by the appearance or transformation of the actors involved in the publishing process and the relations between them (e.g. the use of EDI as the main mechanism for supporting business transactions, direct delivery of materials from publisher to library etc.). However, it is outside the scope of this research as it will not impact on the design and presentation aspects.

3.2 Focus of Research

In the context outlined below, this research concentrates on the analysis of the tools required to support the Author - Electronic Publishing Houses - Library - Users publication channel, as shown by thicker lines in Figure 3.2. More specifically it is concerned with the acquisition and authoring processes required to provide electronic books with an appropriate interface and readers' services. The simplest route, which starts from the input of the original manuscript to the presentation of the final electronic book to the user, has been chosen. Electronic journal publishing would be another interesting subject of investigation, but provides additional features which complicate the channel. The focus of research deals with:

- the definition of a flexible environment for producing electronic books in a semi-automatic way (*hyper-book builder*);
- the definition of a model for electronic books (*hyper-books*) which is suitable for a large range of potential users and does not require computer experience to be understood and used.

The final electronic books are designed with the intention of being used within the context of an electronic library. They also provide a set of features so that they could also be used as stand-alone systems.

There are other systems which share similarities with the features discussed above: for instance book models have been proposed elsewhere (Benest and Duric' 1990, Feldman 1990, Barker and Manji 1991, Barker 1992, etc.); other systems exist which support the conversion of SGML documents into electronic books but which are different from the model proposed by this research, such as Dynatext (Electronic Book Technologies 1990); experiments in the creation of electronic libraries also exist, but most of these are small scale projects (e.g. Bogaerts and Agema 1992) or, where they are larger scale, use different representations for their electronic texts, e.g. Project Gutenberg (Hart 1992), Core (Lesk 1991), Mercury (Arms and Michalack 1990) and Elinor (Wu et al. 1993).

However, this research proposes an innovative system which incorporates the following features: a model of an electronic book based on the book metaphor and designed with the explicit intention of being part of an electronic library; and an environment which supports the semi-automatic creation of electronic books. It also proposes a method for evaluating the resulting electronic books.

3.3 Implications from the Literature Review

In the previous chapter a number of topics have been discussed: paper books and paper publishing, electronic books, electronic publishing, and electronic libraries. In the following, the different topics will be briefly discussed in order to highlight the implications for this research.

(a) The study of paper books and the analysis of design principles involved in paper publishing (typographical conventions, readability problems, etc.) have been taken into account in the design of an electronic publishing environment.

The main conclusions drawn from the paper book study are that few typographical rules are universal: factors such as the purpose of the document, the cultural background and the school of typography determine the rules which will be applied. However, a number of general conventions are usually considered. They concern for instance the choice of the typeface, the choice of the page components, the way text is emphasised, the way text is paginated, etc. The *hyper-book builder* takes into account some of these conventions, even if the electronic medium introduces some implications which need to be taken into account. This is described in more detail in section 3.4.

(b) Research in the field of electronic books (see section 2.3) has shown that there are different interpretations of the concept of the electronic book. In particular, electronic books may be classified into three main categories according to the presentation style:

- portable electronic books;
- electronic books which preserve only the logical structure of paper books;
- electronic books which support both the logical and physical aspects of paper books.

The category which was considered most appropriate - taking into account the context (library), the wide range of potential users, and the kind of publications (scientific books) - is the last one. In fact, both logical structures and physical clues (e.g. pages, book thickness) are considered important for recognising an object as a book, thus guaranteeing ease of use and assimilation of its functions.

It is important to highlight that the choice of maintaining a high similarity with the paper book aims to involve users as much as possible in the employment of electronic capabilities and tools when they could provide advantages not available using the traditional tools. Once people are used to such a system, it will be possible to move gradually towards more and more powerful, flexible and dynamic systems. The result of this evolution could go far from the initial model, and the book metaphor could be partially or completely abandoned in future. However, what is important to underline here is that the starting point should be an object which users know and with which they are familiar. A typical example of a system which has not being widely used is the Teletext; although it provides a large range of up to date information, the access is quite slow (through a classical table of contents) and the interface design is very poor.

(c) Research in the field of electronic publishing has shown that, while traditional paper publishing is governed by processes which have long been established, electronic publishing is still at an early stage. Three steps are involved in the electronic publishing process (acquisition, processing and delivery) as described in section 2.4.2.

Markup languages have a fundamental role in the electronic publishing process, mainly in the acquisition step. The study of markup languages has been necessary in order to know how electronic documents are marked and to choose a language which was appropriate for the source texts in the process of producing electronic books (see next section for more details).

The study of markup languages has led to the choice of SGML (Standard Generalised Markup Language) because it offers a number of advantages over other

markup techniques (see section 2.4.3 in the previous chapter). First of all, SGML is an international standard and is the result of the desire of publishers to have a standard means for preparing documents for publication. The advantage of a standard language is essentially that communication is made easier, and the cost of training skilled staff is reduced as just one system has to be learned, and documents can be re-packaged with relative ease. In addition, SGML guarantees independence both from the system and the application. This represents an additional advantage of using SGML, for instance, when the same document has to be accessed on different machines. Only the logical structure is indicated by the markup; no layout tags are included, therefore the same document can be presented in many different ways according to how the different logical components are formatted. SGML permits the definition of different classes of document. SGML is therefore a meta-language, as it contains the rules for creating an infinite variety of structural markup languages.

The first problem which has been addressed in the design of the *hyper-book builder* was the definition of a structural model for the class of documents which was used in this research (i.e. the definition of a DTD). Different DTDs, which are now in the public domain (e.g. CALS, AAP DTD), were considered in order to decide whether there was an existing DTD suitable for the purpose of this research or whether it was necessary to define a new ad-hoc DTD.

Another interesting aspect related to electronic publishing deals with presentation issues. The kind of document, the kind of users, and the user needs are important factors to be taken into account when designing the interface.

Hypertext is an approach which may be used to represent non-linear texts in a familiar and user friendly way. Some hypertext features (e.g. links, history mechanisms) have been employed in the design of *hyper-books*. On the other hand, some recognised limitations of hypertext, mainly disorientation, have been overcome by incorporating the book metaphor. The user of a book can easily know where (s)he is thanks to some clues such as page numbers, headers, thickness, and how to reach another location (Table of Contents, Index, etc.). Therefore, the *hyper-book* model tries to incorporate the flexibility of the hypertext and overcomes, at the same time, the disorientation problem by using the book metaphor.

Particular attention has been given to the issues related to the conversion of text into hypertext because of the similarities with the process of producing electronic books:

- if the text is available on paper, first of all it is necessary to convert it into an electronic form;

- if the text is already available in electronic form, the kind of markup determines the conversion process; if the text is flat or unmarked, a manual conversion has to be adopted; otherwise an automatic or semi-automatic conversion can be used;
- the definition of a link structure represents an important step both for the hypertext and electronic books;
- indexing techniques can be used as a complement to the browsing facility.

(d) The study of the electronic library has been conducted in order to evaluate the main issues related to the availability of the full-text of documents in electronic form. The library can be accessed remotely from any location through a network using personal or portable computers. The contents of such a library, electronic books, no longer suffer from the constraints of their physical counterparts and can be replicated as many times as required. The library itself can contain an immense amount of books, as a result of the storage capacity of the electronic support and the possibility of accessing other information sources through the network.

A number of artificial intelligence techniques and language based techniques have been implemented in order to automate operations such as cataloguing, classification, search and document retrieval. Although it is outside the scope of this research, it can be anticipated that such operations would form part of the larger Super-Library project.

3.4 Design and Development of a Prototype

This section anticipates the content of the next chapter and summarises the main features of the *hyper-book* and the *hyper-book builder*.

Hyper-book is an alternative approach for representing electronic text which overcomes the problem of limited familiarity with most of the electronic text presentations. As mentioned in the previous section, *hyper-book* incorporates some hypertext features (e.g. history facilities and links), most characteristics of the paper book, and some other features (e.g. searching facilities) which are made possible by the electronic support.

Figure 3.3 sketches the *hyper-book* position as an intersection between paper books, hypertext, and other electronic capabilities which are not included in the two previous models, e.g. searching facilities.

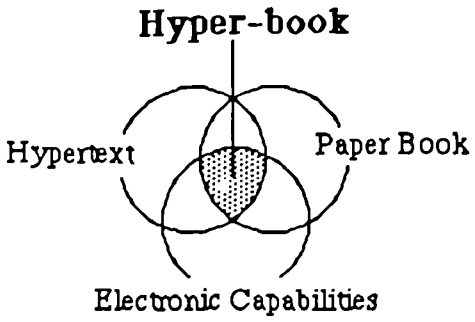


Figure 3.3: The hyper-book position

Design principles for *hyper-books* have been partially deduced from paper publishing and from contemporary experiments with electronic books (Benest and Duric' 1990, Burrill and Ogden 1989, Catenazzi and Argentesi 1991, Barker 1992, etc.). As anticipated in the previous section, the *hyper-book* model takes into account some typographical conventions adopted for paper publishing (e.g. the choice of the page components). However, the flexibility of the *hyper-book builder* allows the publisher to modify the book layout by associating, for instance, new typefaces, or by moving the position of the page components.

If the electronic support permits more flexibility, it also introduces some constraints, such as limited screen size, and low resolution, which have to be considered in, for instance, the definition of the size of the book.

The *hyper-book builder* is the environment within which *hyper-books* are produced. The design of such an environment has been supported by the literature in the field of both paper publishing and typography and the study of existing experiments of conversion of SGML documents into other formats.

The generation process is carried out semi-automatically. First, an electronic text, in SGML or ASCII format, is imported into an *empty template*, creating a first version of the electronic book. Then, the publisher can modify the layout style and decide which services to provide to the reader, producing the final version of the electronic book.

The fact that the full-text of documents is electronically available in a markup form introduces the possibility of exploiting this markup in order to automate some of the operations which are needed in order to insert *hyper-books* in an electronic library such as cataloguing, abstracting, and automatic indexing.

A test application was developed on the basis of the design plan previously defined. The prototype has allowed the demonstration of most of the features of the *hyper-book* model and the *hyper-book builder*, thanks to a powerful and easy developing environment (HyperCard). Two books have been organised according to the *hyper-book* model. The choice of the books has been constrained by the general project, Super-Library, within which this research was conducted.

Chapter 4:
Hyper-book Design

As stated in the previous chapter, this research concentrates on a particular channel of the electronic publishing process, where the main players involved are: the author, who writes and submits his manuscript in electronic form; the publisher, who is responsible for the authoring activities which convert the original manuscript into a specified electronic book format; the library, which acts as a repository of electronic publications; and the user, who actually accesses the electronic library material. Within this context, this research concentrates on the acquisition and authoring processes required to produce an electronic book which provides appropriate text presentation and reader services.

More and more information is now available in electronic form but it is not normally presented in a convenient and user-friendly way. *Hyper-book* is an alternative approach for presenting and making accessible electronic texts in a familiar and user-friendly way. An important feature of such electronic books is the possibility of accessing text in a non-linear way, following the hypertext philosophy. For that reason they are called *hyper-books*[1], and the environment where they are produced, the *hyper-book builder*.

---

[1] The term "hyperbook" has also been used as a name for a Longman/Logotron product which is no longer being marketed. The choice of the name *hyper-book* for the author's electronic book was made before this product was launched. It has been retained as it seems the most appropriate term for electronic books in which interconnection is one of the main features.

The *hyper-book* model is defined on the basis of the paper book metaphor. The paper book model is enriched with several tools which are made possible by the electronic environment (history facilities, searching facilities, links, etc.). The *hyper-book* supports several book control functions (turn page, close book, etc.) and a number of reader tools (orientation, personalisation, searching, history, navigation and printing facilities), all of which are described below. A detailed description of how the book metaphor is interpreted in *hyper-books* is reported in section 4.4.

The *hyper-book builder* is the environment within which *hyper-books* are produced from a marked text. A number of tools have been provided in order to allow the publisher to associate his own layout style as well as to decide which reader services to provide to the reader, taking into account the characteristics of the reader and the application domain.

Design principles for electronic books, considered in terms of types of interface, styles of interaction and facilities and services provided to the users, may be partially deduced from traditional book design principles and from contemporary experiments with electronic books (Benest and Duric' 1990, Barker 1991, Barker and Manji 1991, Barker 1992, Catenazzi and Argentesi 1991, etc.). Many aspects of paper book design are also valid for electronic books although some constraints (such as limited screen size, low resolution) on the one hand, and more capabilities on the other (flexibility, interactivity), contribute to creating a very different object. The size of the book, for instance, is constrained by the screen dimensions. The resolution of the screen also means that the font size needs to be bigger on screen than on paper in order to be legible.

Another factor which influences electronic book design is the nature of the end-user and the purpose for which the book is intended. The *hyper-book* model cannot be adopted for all kinds of electronic publication (see below). It is proposed here as a model for scientific books which exist within an electronic library and some properties are dependent on this feature. In a library, users typically find a book after a bibliographic search or browsing through the library shelves. They then consult it in order to see if it is what they are looking for, and, if the book seems to be interesting, users borrow it. At this point a more detailed study will be carried out. Most of the *hyper-book* facilities support this kind of approach: browsing, followed by personalisation and finally printing of the document for more detailed study. The *hyper-book* model is not considered valid for every kind of electronic material which

is also available in the library: online catalogues, for instance, cannot benefit from the *hyper-book* model as users are familiar with a completely different model which provides different access keys to the bibliographical material and does not consider the book metaphor at all.

Although outside the scope of this research it should be noted that - as the full-text of documents is electronically available in a marked up form - there is a possibility of exploiting these marked up texts in order to automate some of the operations which are needed in order to insert *hyper-books* in an electronic library, such as cataloguing, abstracting, and automatic indexing.

Firstly, this chapter describes how *hyper-books* are generated in the *hyper-book builder* environment. Secondly, the *hyper-book* is presented and formally described as a dynamic system. The potential of *hyper-books* in the context of the electronic library is then described. Finally, the *hyper-book* model is compared to the paper book from the following points of view: task, semantic, lexical and physical. The last point of view introduces a number of features which will be fully described in the next chapter, the *hyper-book* implementation.

4.1 Hyper-book Builder

The purpose of the *hyper-book builder* is to provide an environment for creating *hyper-books* through a set of tools which are described below. It is not necessary for the purpose of this research to consider the various players involved during the preparation of a book (editors, typographers, designers, etc.). It is assumed that the notion of publisher subsumes most of these roles. The text is assumed to be in its final version. It is worth noting that all the activities (mainly the editorial process) which lead to the final version can be significantly facilitated by the fact the text can be exchanged in a standard format.

The *hyper-book builder* is an environment in which *hyper-books* are produced. Two activities are basically supported: definition of the *hyper-book* interface; and interpretation of the markup and presentation of electronic texts according to that interface. The *hyper-book builder* is meant to provide the tools for creating *hyper-books* semi-automatically; a publisher, having defined a general logical model for the book, asks the authors to submit their manuscript according to that model. The model has been translated formally in terms of the SGML markup language. If authors

cannot provide their manuscript in such a format, an intermediary will be responsible for converting them into the required format. If the text is available in a marked format, it could be automatically converted into SGML format using translator programs. If the text is unstructured (ASCII), it can be either imported directly into the *hyper-book* format, without any structural information, or can be manually marked up using commercially available editors and taggers (see the SGML section in the literature review).

A *hyper-book* is generated starting from empty templates and a text that is in SGML or ASCII form. A template defines a model of a book in terms of the logical structure (i.e. in terms of the logical components of the book: chapters, sections, etc.), the format (that is layout or presentation style) and possible reader services. A number of predefined templates are provided. A set of editor tools enable the publisher to modify some aspects of the book format and also some reader services (see section 4.1.6), dependent upon the reader's needs and the application domain. In order to generate *hyper-books*, these empty templates are filled with the text. The book can then be further modified.

The screen space of the *hyper-book builder* is divided into three main areas: the *editor tools* area, the *reader tools* area and the *book space*. The editor tools area contains the set of capabilities the publisher is provided with for modifying the book format and selecting appropriate services for the reader. The reader tools area contains the services provided to the reader when the *hyper-book* is completed. The book space, which occupies the largest part of the screen is the area dedicated to the book itself.

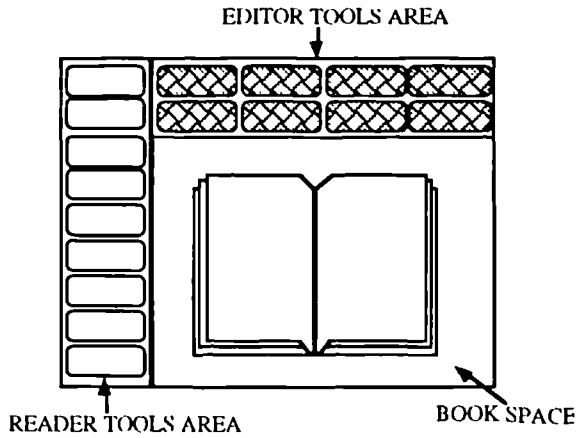


Figure 4.1: The screen layout

4.1.1 SGML and the Hyper-book Document Type Definition

The purpose of the *hyper-book builder*, i.e. the automatic translation of electronic documents into the *hyper-book* format, can be achieved if the source text contains a proper markup. SGML (Standard Generalized Markup Language) offers a number of advantages over other markup techniques, as described in the literature review chapter; therefore, it has been chosen as a markup language for the source documents.

The kind of documents considered in this research are scientific publications, and in particular scientific books. They are usually well-structured documents; they contain text as well as graphics (diagrams, pictures, photos, tables); they provide different access mechanisms (Tables of Contents, Index); they contain frequent references within the same document and even to other documents; and they allow a random access mode of reading.

The first problem to be addressed was the definition of a structural model for such a class of documents which is general enough to include different instances within the same class. The model has since been formally expressed through a Document Type Definition. The original idea was to use an existing DTD as many DTDs are now in the public domain (CALS, AAP DTD, etc.). The British Library SGML starter set (Smith 1987, Smith 1987a) was chosen as a starting point because it is sufficiently detailed and complete for the purpose of this research, and at the same time is quite simple when compared, for instance, to the AAP DTD which has something like two hundred tags many of which are meaningless to non-Americans (Smith 1987a).

For reasons of simplicity the British Library SGML starter set has been further reduced. The purpose of the project focuses more on demonstrating the justifiability and utility of such a system rather than on developing a commercially acceptable product. Some *elements* used in the British Library SGML starter set have been omitted and most of the *entities* of the starter set have not been included. The resulting DTD is called the *hyper-book DTD* and is presented in Appendix A. The selection of the elements in the hyper-*book DTD* has been determined by the analysis of a sample of documents used as the test application. Other elements could be introduced, but those selected here are considered to be both sufficient and representative for the purpose of this research.

According to the *hyper-book DTD*, a book consists of three main parts: the front matter, the main body and the back matter, as shown in figure 4.2.[2]

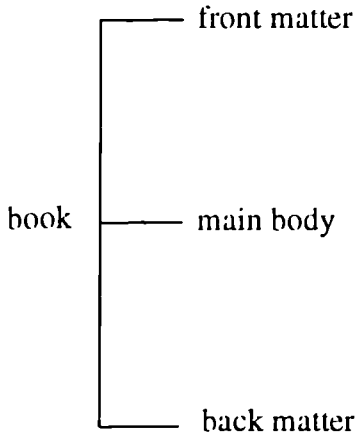


Figure 4.2: The structure of the book

In the British Library SGML starter set, the book element contains an additional part: the appendix. In the *hyper-book DTD*, the appendix has been included as a subelement of the back matter element.

The front matter may include elements such as the title page, the verso page, and a number of other elements such as the abstract, foreword, etc. Such elements may occur in any order and some of them are optional. The character "?" at the end of an element name in the diagrams indicates that the element may occur 0 or 1 time. The character "*" indicates that the element may occur 0 or more times. The character "+" indicates that the element may occur 1 or more times. The fact that some elements may occur in any order, according to the *hyper-book DTD*, is not expressly indicated in the diagrams.

The title page may contain the title, sub-title, author(s), editor(s), the publisher, corporate body(ies), place(s) of publication, date of publication, information about the series and edition, and other material which has not been included in any element previously mentioned.

The verso page may contain details about the edition, the publisher, the printer, the printing, and the copyright; it may also contain the CIP (Cataloguing In Publication), the ISSN (International Standard Serial Number), the ISBN (International Standard

[2] Figures 4.2-4.7 are graphical representations of the structure of the book expressed in terms of its components; the attributes have been omitted for reasons of simplicity. These figures are not supposed to be a formal translation of the *hyper-book DTD*.

Book Number) and other material which has not been included in any element previously mentioned.

Other elements which may occur in the front matter are the abstract, foreword, preface, acknowledgements, dedication, Table of Contents, List of Tables and List of Figures. The last three are defined as empty elements, i.e. elements without contents; their contents will be automatically generated during the *hyper-book* generation process. Figure 4.3 illustrates the structure of the front matter.

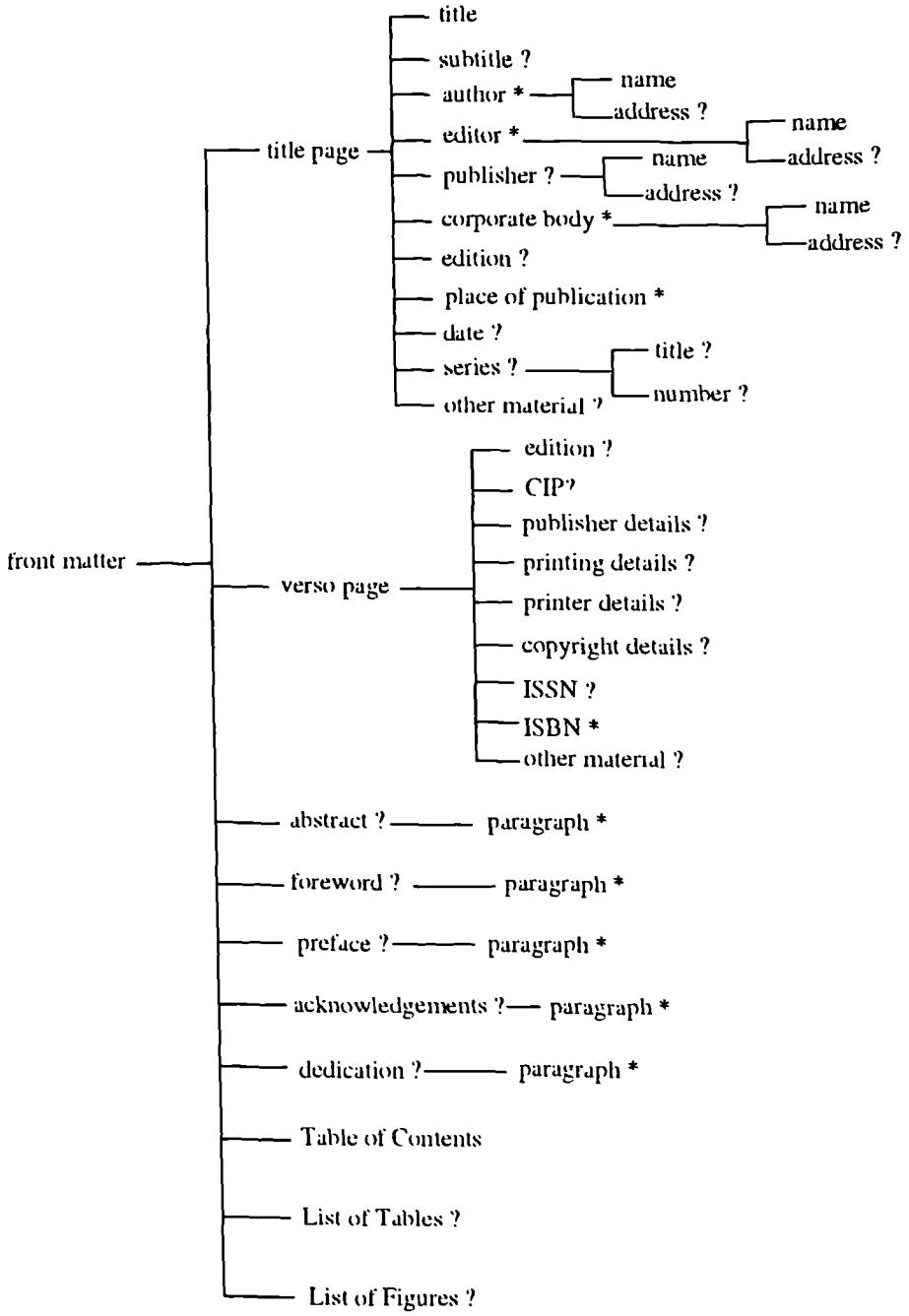


Figure 4.3: The structure of the front matter

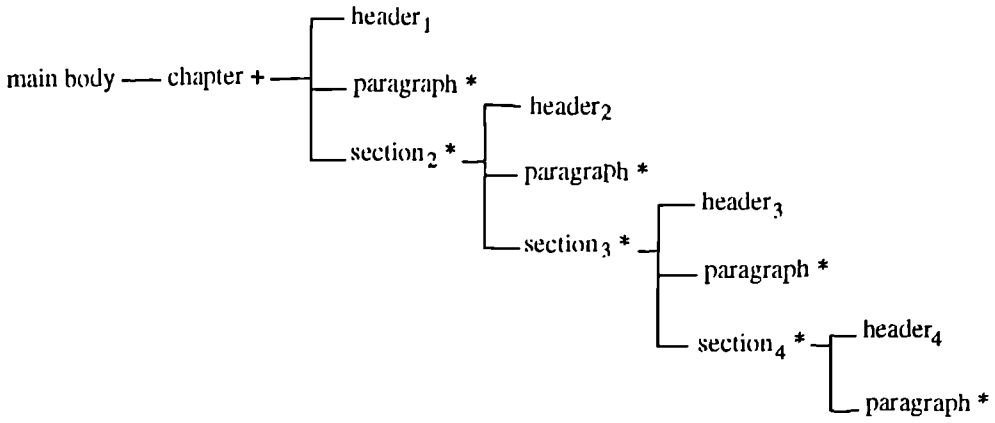


Figure 4.4: The structure of the main body

The main body is divided into chapters (first level sections), each of which is composed of sections and each section of subsections, down to four levels (as shown in figure 4.4). This could be further expanded, but four levels are considered sufficient for the purposes of this research. Each section has been given a heading, and, optionally, some text up to the next logical division of the text as shown in the following example.

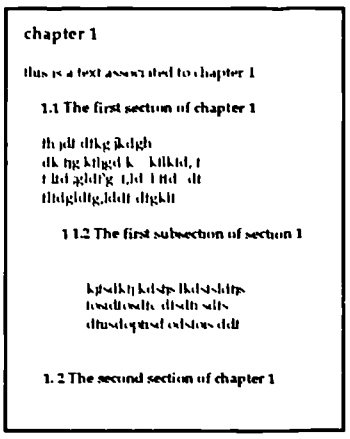


Figure 4.5: An example of the structure of the main body

Chapters may contain graphics and footnotes. They are called floating objects as they may occur anywhere (in the main body) and their position is usually defined once the marked document has been formatted. These are not explicitly indicated in figure 4.4 even if they are catered for in the system, as indicated in Appendix A. Graphics

include all kinds of material which is not text, in particular figures and tables; they are optionally given a caption. Information about graphics, is expressed as attributes.

Particular attention has to be paid to what has been generally defined as text. The basic elements of text are paragraphs which consist of characters but may contain links and index entries, which allow the topic index to be automatically generated[3]. The topic index will contain entries at two different levels. This is a simplified version of the British Library SGML starter set, where three level entries are foreseen. For each second level index entry, there is a parent first level index entry. This information is inserted as a value of the *parent* attribute. A topic index entry may have a *see also* attribute which indicates related entries. The following example will clarify this concept showing an index entry generated by *hypermedia* and *hypertext* as first level index entries, *application* and *authoring* as second level entries, whose *parent* attribute is *hypertext*. *Hypertext* has a *see also* attribute, whose value is *hypermedia*.

hypermedia, 56

...

hypertext (see also hypermedia), 49

application, 89, 90

authoring, 78

Other types of index may occur in a book such as author index and numeric indexes. They have not been included in the *hyper-book DTD* as they are not relevant for the test application.

Links include different types (links to graphics, footnotes, bibliographical items, etc.) which correspond exactly to the link typology described later in section 4.1.4. The link mechanism is based on two attributes, ID and IDREF, which allow respectively the association of an unique identifier to an element and the reference to the value of an unique identifier for creating the connection. The arc connecting the segments, in figure 4.6, indicates an "OR" relation between elements at the end of the segment.

[3] An index produced in this way is called a *derived index*, because it is automatically generated from terms in the text; this is in contrast to the concept of an *assigned index*, where an expert creates the index by identifying the set of index terms appropriate for a book.

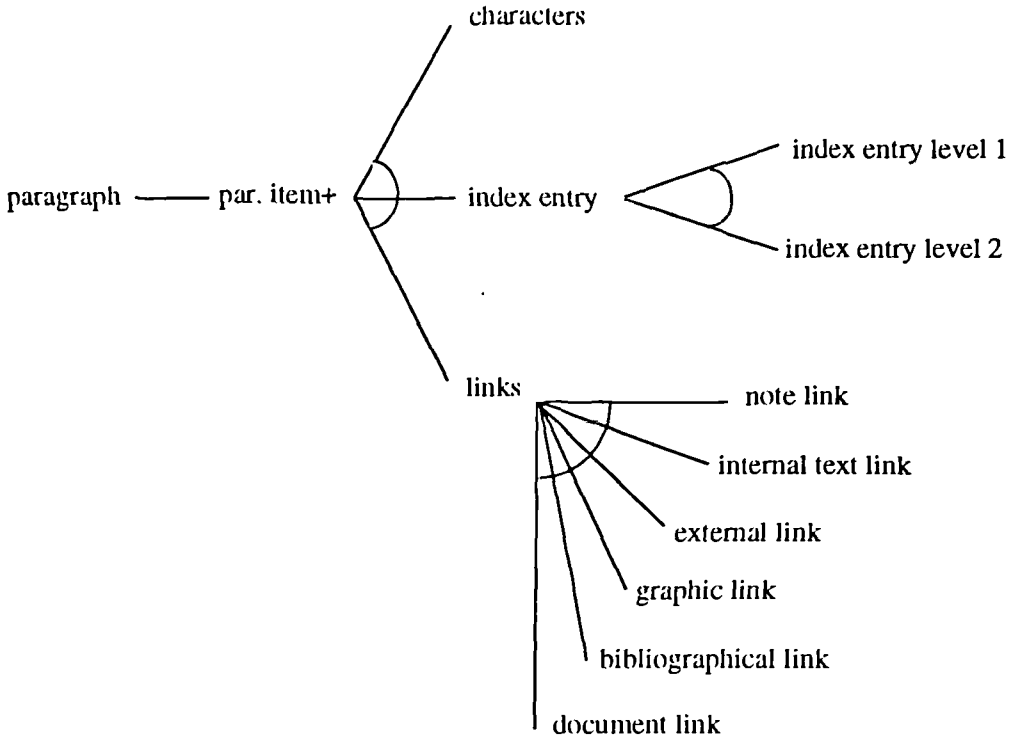


Figure 4.6: The structure of the text

The back matter (see figure 4.7) may be composed of references, a topic index, a glossary, a list of related documents (for instance books which belong to the same series), acronyms, biographical material, and appendices which have the same structure as the main body structure.

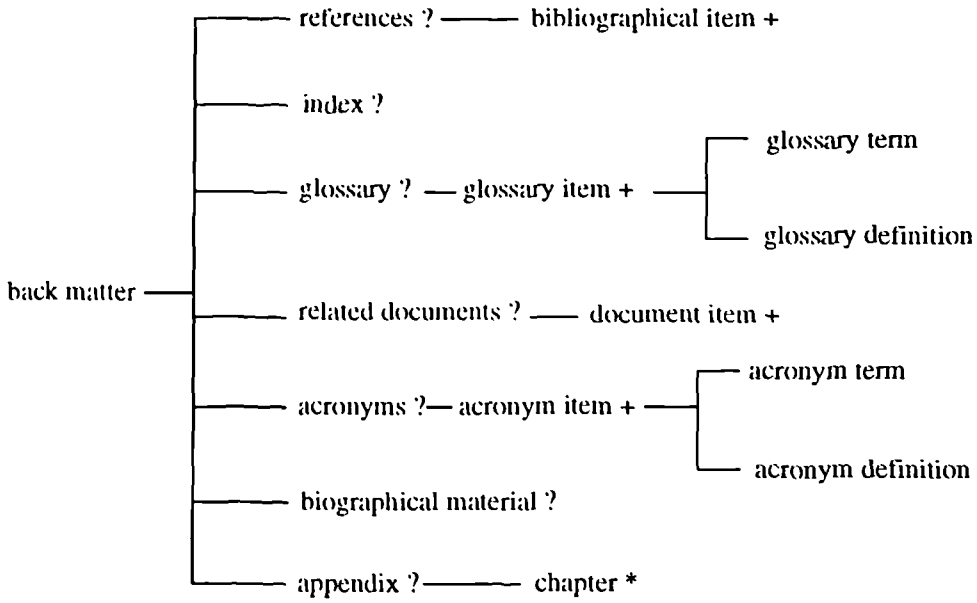


Figure 4.7. The structure of the back matter

4.1.2 The Empty Template

The study of paper book design issues has been particularly important in designing empty templates in the *hyper-book builder*. A template is a model of a *hyper-book*; it represents it in terms of structure, format and reader services. The contents are obviously not included. The term *structure* is always used to refer to the logical structure and is determined by the DTD if the input text is in SGML; no structure is defined for ASCII texts. At the moment, the only DTD considered is the *hyper-book DTD*; however, it would be possible to define empty templates for other DTDs.

The format is relative to the presentation of the text and to its appearance on the screen. Paper book design principles have been mainly employed in the definition of the format which specifies a number of layout features:

- book size;
- page size;
- margins;
- typefaces, style and size to be used in different parts of the book;
- design of the page and its components;
- design of specific pages (Title page, Verso page, etc.).

A *hyper-book* is presented as an open book with two pages side by side in view. A page typically contains the header, the footer and the text. Special pages such as Title and Verso page contain the elements mentioned in the *hyper-book DTD*.

The reader services include a number of facilities (personalisation, searching, orientation, etc.) which, during the book generation phase, may be defined by the book publisher for the reader.

For a given DTD, it is possible to define a number of different templates with the same logical structure, but different formats and reader services.

The empty template also includes the editor tools, described in detail in section 4.1.6, which allow the publisher to modify the book format and reader services before the *hyper-book* generation process is activated. For instance, it is convenient to predefine the appropriate book and page size in the empty template in order to avoid successive changes and consequent repaginations when the book is already full.

4.1.3 Hyper-book Generation Steps

Different steps are involved in the process of generating *hyper-books* (see figure 4.8). The first one consists of the automatic translation of SGML texts into the *hyper-book* format. A template will be chosen by the publisher from the set of templates valid for the *hyper-book DTD* before starting the process. The document is first parsed to verify that it is conformant with the *hyper-book DTD*, then some translation rules are applied which map the different elements of the document into a format specified by the template. In the main, the sequence of elements in the original document is maintained during this phase; the text of the main body is paginated (see section 4.1.5); special styles are associated with special parts of the text: section headers, glossary terms, link sources. Link information is extracted and saved in special structures while the link sources are made active. Table of Contents and index are automatically generated as well as headers and footers. The Table of Contents is generated starting from the headings of the different sections and subsections defined in the book. The index is generated starting from the index terms marked in the text which are then extracted, sorted and linked with the corresponding part in the text.

If the text is available in ASCII form, the generation process is reduced to simply paginating text in a template.

The next step consists of manipulating the resulting book in order to improve its appearance and make it available to the reader. The changes deal with the *hyper-book* format and the reader services: the publisher is free to modify the format, for instance, by associating a different font to some part of the book or by changing the page size. The *hyper-book* format and the reader services may also be changed directly on the empty template before the *hyper-book* is actually generated.

The book resulting from the translation step contains links, which are directly derived from the original SGML text, provided by the author. The publisher is provided here with the capability to modify the link structure by adding new links (see *link update* in section 4.1.6).

In order to complete the generation process, another functionality, indexing, may be provided. The form of indexing supported is full-text. There are several reasons which justify this choice (Kirstein 1993): firstly, people often exactly remember a word in a text, for instance a date or a name, and they need a mechanism for finding that word; secondly, the cost is relatively low, compared to keyword indexing, and the indexing process can be automatically implemented.

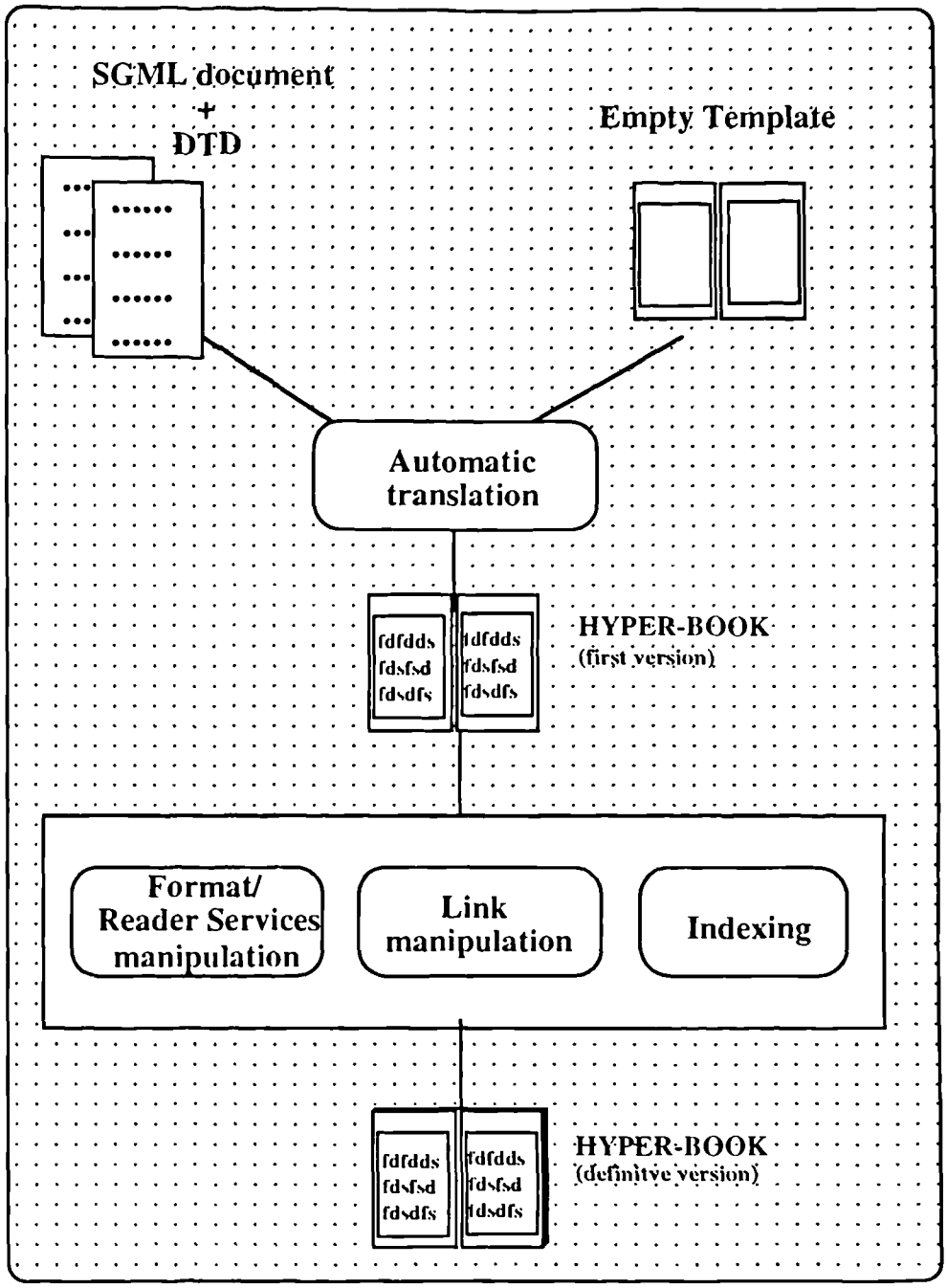


Figure 4.8: The hyper-book generation process

Once the different book manipulation processes have been completed, the *hyper-book* is ready for the reader. All the editor tools are removed, because the reader is not allowed to change the text of the book, and the reader services are made available.

It is important to note that when the author first submits his manuscript in SGML form, the full details of the title and verso page cannot be inserted. The publisher is

responsible for completing the document with the information which is missing before activating the *hyper-book* generation process.

4.1.4 Link Definition

Hyper-book links are based on logical addresses (e.g. section identifiers) rather than on physical ones (page numbers). This allows the dynamic nature of *hyper-books* to be catered for during the building phase, e.g. the fact that page numbers can change every time the book is repaginated while logical addresses are uniquely defined at the creation time.

A *hyper-book* link is described in terms of type, link source and link destination. Different types have been defined:

- *internal text link* (xi): a link to a structural element of the *hyper-book* such as a chapter, a section, the foreword, the index;
- *note link* (n): a link to a footnote;
- *graphic link* (g): a link to a graphic object such as a figure or a table;
- *bibliographical link* (b): a link to an item in the references;
- *document link* (d): a link to an item in the list of related documents, which appears in the back matter;
- *external link* (xe): a link to an external book or to its structural elements.

Links to acronyms are not explicitly defined but acronym terms in the text are made automatically active when the SGML document is translated into the *hyper-book*.

The link source is an active string or word in the text. The link destination is the logical object referred to by the link and depends on the link type; the logical object is specified by means of its unique identifier. As described in section 4.2.1, the effect of the link activation can be in a jump to the referred object or can simply cause the display of an overlapped window which shows the required information.

The basic link structure is automatically defined during the generation process by interpreting the link markers in the original text. Links described so far (transverse) are all defined by explicit references in the original text. Other links (hierarchical) are defined in the Tables of Contents, List of Figures, List of Tables and Index.

4.1.5 The Pagination Process

Pagination is defined as the process of laying out the parts of a document into pages. Pagination is a very complex process which should take into account a number

of issues as described in the literature review chapter. *Hyper-book* does not pretend to offer the functions of a document formatter or a sophisticated word-processor. However some basic rules have been applied during the pagination process. A page break is introduced before first level headers, i.e. every new chapter starts at a new page. A basic treatment for orphan lines is available when orphans are represented by section headers. Words are not hyphenated across a page break. Footnotes do not affect the pagination process because they appear if required in an overlapped window in the same page where the footnote reference appears. Graphics require special treatment. If the space available in the current page is not enough to contain a picture, the picture is moved to the next page; again if the space is too small, the picture is automatically scaled in order to fit the space available. A zoom facility allows pictures to be shown in their initial size and solves the problem of poor legibility due to the reduced size.

4.1.6 The Editor Tools

A number of editor tools are used both before and during the building process in order to change and/or improve the physical appearance of the *hyper-book* which is automatically produced during the first steps. Their purpose is to allow the definition of an object which responds to the needs of a group of readers with specific tasks. In other words, they are tools for customising a book both in terms of the format and reader services.

The editor tools include:

- *change font*: allows the changing of typeface, style and size of any portion of text from the page. When the book is first generated a number of predefined text formats, i.e. information about the typeface, size and style, are automatically applied to different elements of the text such as section headers, figure caption, etc. Such text formats may be modified and reapplied.

- *change size*: allows the changing of the size of the book, the page and its components, e.g. the title and author fields in the title page, footer and header in the main body of the text. A change performed on any element of the right or left page (the header, the footer, the text) symmetrically affects the corresponding element, if there is one, on the left or right page respectively. All the pages in the book are affected by the change. Exceptions are the cover page or the title page, where the elements are unique

to that specific page. Obviously, the book components are constrained by the size of the book and the book itself is limited by the book space.

- *move objects*: allows the changing of the position of the book in the book space and the position of its components within the book borders. A change performed on any element of the right or left page (the header, the footer, the text) symmetrically affects the corresponding element on the left or right page respectively. All the pages in the book are affected by the change with the exceptions mentioned for the change size tool.

- *link update*: the book publisher is allowed to expand the existing link structure with new links which are considered useful; as *hyper-books* have been conceived as part of a collection (the library), new links are highly likely. This tool is a means for adapting the *hyper-book* for the environment where it has been placed (i.e. creating appropriate connections) and is not meant for altering the original author's intention.

- *selection of reader services*: the editor may decide which services to provide to the reader and how to provide them. A number of reader tools which could be useful in general are already provided and could be removed by the publisher if they are considered unnecessary: annotation, bookmarks, searching, printing facilities, etc. Other services could be added by the publisher taking into account the specific application, the particular group of readers and their tasks (for instance chemistry books can benefit from programs which can compute chemical formulas; other books can be extended by integrating multimedia components, etc.). In general, incorporating new services is a not an easy problem and is strongly dependent both on the application and implementation issues.

- *end work*: concludes the *hyper-book* generation process and makes the *hyper-book* ready for the reader. All the editor tools are removed and the book cannot be further modified except by using the reader services which are now available. The editor tools area is now replaced by a library browser which indicates the *hyper-book* position in the library and allows the user to navigate in the library itself. The browser can be very simple if the collection is limited, for instance a book shelf, or more complex, for instance a subject network, for large collections of books.

In addition, the book publisher is provided with some of the tools which will be described as reader services and which are always available: navigation, orientation, history tools (see section 4.2.2).

4.2 Hyper-book

Hyper-books maintain many of the features of paper books and provide added-value that exploits the technology underlying it, such as indexing, links, and history mechanisms. The *hyper-book* representation, content, and reader services are described in detail in the next sections. A formal definition of the *hyper-book* model as a dynamic system is given in section 4.2.3.

4.2.1 Book Representation and Content

A *hyper-book* can be presented closed, with the front or back cover in view, or open. In accordance with Benest's model, when the *hyper-book* is open, two pages are displayed simultaneously and the position within the text is clearly indicated by the thickness of the pages at the two sides. The logical book structure is determined by the chosen DTD (*hyper-book DTD*) while the format is determined during the building process.

The *hyper-book* content is dependent on the content of the original electronic document, which consists of text and graphics. Graphical objects, always visible, appear in the text as in a paper book; this represents an additional orientation clue when the book is consulted. This solution is in contrast with another frequently adopted approach (Egan et al. 1989, Electronic Book Technologies 1990) of making pictures available in an overlapped window as and when they are required. In addition they provide a zoom property which allows them to be seen in their original size if they have been reduced during the building phase (see pagination process). Another feature of the text is that it contains embedded links.

From the reader's point of view, *hyper-book* links can be grouped into two main categories: hierarchical links, which are defined in the Table of Contents, List of Figures, List of the Tables, and Index, and transverse links, which are defined in the text. Transverse links may be divided into two other groups: window links and jump links. The first group (window links) consists of links whose activation causes the display of additional information in an overlapped window (explanatory window); the

second (jump links) consists of links whose activation allows access to cross-referenced information in another part of the same book or in another book.

The first group includes links to footnotes, links to bibliographical items or to items in the list of related documents, and links to acronyms. For this kind of link, a jump to the relative section could be distracting, as what is required is just that piece of information.

Links to structural elements of the book (sections, subsections, etc.), links to graphics, links to external documents or to their structural elements, belong to the second group (jump links). In this case it seems more effective to move directly to the piece of information referred to by the link (e.g. a figure and a subsection), maintaining the visual and semantic context within which this information is located. In this case a backtracking mechanism allows the reader to go back to the page where the link has been activated.

Links are highlighted in the text using a special font style. Such style can be suppressed, if it is found to be a distraction during the reading process. In addition, in order to distinguish the different kinds of links, different solutions are used: for instance, links to bibliographical items can be indicated by strings in square brackets representing a bibliographical code (e.g. author's name and date of publication). Usually, the link source is in itself a clue for recognising the kind of link; for instance, "see table 2" is clearly a link to a table while "in chapter 3" is a link to a structural element of the book.

4.2.2 Reader Services

The reader services are the tools provided to the reader in order to improve his/her interaction with the *hyper-book* They include:

- Navigation tools

There are several ways of navigating in a book: the main are hierarchically (e.g. from the Table of Contents down to the units of text) and transversally (references across the text). *Hyper-book* provides both of these methods. Transverse links include internal links (reference across the document) and external links (links across a library). Another way of moving in the text is the classical linear way. *Hyper-book* offers the possibility of going backward and forward through the pages of the book as well as flipping through pages. In addition, direct access is provided to a number of

special locations: the Table of Contents, the List of Figures, the List of Tables, the index, the front cover, and the back cover.

As the *hyper-book* is conceived as being in a library, there is also the possibility of accessing other connected books. Those publications which are related in some way to the book being consulted are made accessible. The type of relation may be defined using different criteria: books by the same author, in the same series, with the same subject, etc. This facility is mentioned here to indicate the potential of *hyper-books* in the electronic library, but it is outside the scope of this research to investigate in detail the related issues.

- Orientation tools

The main method of providing orientation in a paper book is the Table of Contents. In addition, paper books use running heads and the thickness of the paper as orientation clues.

A recognised problem for most online texts is the homogeneity problem (Nielsen 1990a), i.e. the fact that most on-line texts always look the same making it difficult for users to get a sense of location. Different solutions have been adopted for providing orientation in an electronic text.

For example, some electronic books show the Table of Contents at all times (Egan et al. 1991, Catenazzi and Argentesi 1991) with a marker on the current position. Others show the Tables of Contents only in the preliminary pages of the book (Benest and Duric' 1990).

Other approaches are adopted in electronic books for indicating the user's position within the book, such as closed pages (Burrill and Ogden 1989) and black bands (Benest and Duric' 1990).

Hyper-books provide a Table of Contents only in the front matter with the possibility of jumping there from any part of the book. In addition, they provide black bands as well as running heads. A black band simulates the thickness of the book. It is composed of vertical and horizontal lines, along the bottom and outer sides of the book, proportional to the number of pages currently opened. It is possible to make large jumps in the book by selecting the black band in a specific place. Thus, the black band is also a navigation tool.

- History tools

When consulting electronic books, it may be useful to remember the path followed during the reading process and to be able to go back to a previously visited location. *Hyper-books* offer such a history mechanism, by providing a tool which lets the user go one-step back or select one location from a set of visited locations.

- Personalisation tools

Hyper-books provide three kinds of personalisation: annotation, notes and bookmarking. These have the advantage that they may be superimposed upon the page without altering the original copy of the book for another user. In an electronic library environment this is perfectly reasonable or even indispensable: a reader is not allowed to modify the library material, however this mechanism allows the reader to customize the book, i.e. to create a personal view.

The kinds of annotations which are provided by *hyper-books* are essentially typing and graphical annotations. The reader should be able to type annotations, highlight text and draw lines on the pages which are currently open. Such annotations overlap the content of the page and are removed using an eraser which does not affect the text underneath.

Another way of adding personal comments is to write notes in an overlapped window associated with the text. Each window has a name, a position on the page and can be moved or hidden if it obscures useful parts of the text underneath. This facility has been included in order to expand the area dedicated to comments when the space on the page is not enough and too many overlapped annotations could make the content of the page difficult to read. This represents an opportunity which does not have a counterpart in paper books, but could have important implications for an electronic book. It could be useful, for instance, to create a note manager, which allows access to a particular note from an active list which shows the names of all the user notes, or to find notes which contain specified search terms.

Bookmarks represent an additional personalisation aid which is useful in order to mark relevant pages. Their peculiar feature is the fact that each bookmark contains an indicator (for instance a page number, a name, or a short note associated with it)

which permits it to be distinguished from the others. Furthermore, bookmarks are active objects and provide an additional history mechanism.

- Searching tools

The Index and Table of Contents represent the two main methods of finding useful material in a paper book. *Hyper-book* provides both of these and in addition it provides full-text search. The result of a search using full-text indexing is visible in the main body of text through highlighted search terms, as well as in the Table of Contents in terms of the search term frequencies for each entry. This approach guarantees that contextual information is always available. Another possibility could be to rank the Table of Contents entries according to the search term frequencies.

- External sources tools

A number of tools make use of resources which are outside the *hyper-book* system, such as word processors, printers, etc.

The reader may copy pieces of text and save them in a personal clipboard for future use. The reader is provided with the capability of opening a word processor, where she/he can paste text which has previously been copied or can write her/his own annotations.

The whole book or selected parts of it may be printed. All the typographical information of the *hyper-book* is preserved in the paper copy. This facility is mainly necessary when large parts of the book must be read. Currently the low screen resolution is still a major limitation when electronic text has to be read from a screen (McKnight et al. 1991, Shneiderman 1992).

4.2.3 The Hyper-book Model

The objective of this section is to give a formal definition of the *hyper-book* model as a dynamic system.

Formal models already exist which describe the structure of electronic books (Barker 1991, Barker and Manij 1991). However, formal models which describe the functional aspects of electronic books do not exist, though they exist in related fields such as hypertext (Tompa 1989, Stotts and Furuta 1989, etc.). Functional models of hypertext are based on the classical definition of hypertext structures (nodes and links)

rather than on the concept of pages and page components, which characterise an electronic book.

The functional aspect is indispensable for describing the use of a dynamic and interactive system. To this end, it is necessary to use the concept of state, i.e. a record of some information that unequivocally characterises the system from a certain viewpoint, and the concept of transition, i.e. an evolution from one state to another (Zadeh and Polak 1969). This transition is accomplished by means of a set of operators, which correspond to the reader services described above.

In the following section, the *hyper-book* model is defined in terms of structural and functional components. Moreover, how the system evolves under the effects of these functional components is presented.

4.2.3.1 Structural Components

A *hyper-book* basically consists of a front cover, a sequence of double pages, and a back cover.

A double page consists of two pages, the left and right page. Each page can contain a number of components of different types: figures, tables, and text.

Up to this point this represents a general definition for describing a book. *Hyper-books* provide additional features, which rely on their electronic nature. Among these features, it is relevant to note the distinction between active and non-active text. An active text is an active area of the page which represents a source for a link. There are different types of active texts according the nature of the link as mentioned in section 4.2.1: <h-text>, <w-text>, <j-text>. They respectively correspond to hierarchical links, window links and jump links, and will be described as operators later in this section.

The *hyper-book* components highlighted above can be summarised according to the BNF notation[4] by the following definitions:

<hyper-book>::= <front-cover> <double-page>* <back-cover>
<double-page>::=<page> <page>
<page>::=<page-component>*
<page-component>::= <text> | <table> | <figure>
<text>::=<active-text> | <simple-text>
<active-text>::= <j-text> | <w-text> | <h-text>

[4] According to BNF (Backus Naur Form), <elem> indicates an element, the symbol "::=" means "is defined as"; <elem>* means 0 or more occurences of <elem>; the symbol "| " indicates a disjunction (OR) of elements.

<front-cover>, <back-cover>, <table>, <figure> are terminal elements. They simply represent the front cover, the back cover, a table and a figure of a book. Other terminal elements are <j-text>, <w-text>, and <h-text>. They represent sources of links as mentioned above. A final terminal element <simple-text> indicates any text which is not <active-text>. All these types of text basically consist of a sequence of characters.

4.2.3.2 Functional Components

So far, the *hyper-book* structure has been described on the basis of which a static model has emerged. A further step in modelling the *hyper-book* system is a formalisation of its functional behaviour. This step allows the dynamic nature of the *hyper-book* system to be represented. The system transitions correspond to the user interactions with the system (e.g. navigation, personalisation).

The notation used so far, is not appropriate for describing the dynamic nature of the *hyper-book* system which depends on its actual use. Therefore, it is necessary to adopt another formalism based on sets, and operators from Systems Theory (Zadeh and Polak 1969).

First, some definitions of sets and concepts useful in the formalisation will be introduced.

The Position

Having provided the basic components of a *hyper-book*, how a *hyper-book* would look is now considered.

A *hyper-book* may be presented closed, with the front cover or the back cover in view, or open, with two pages in view. Therefore, the front cover, the back cover, and any double page correspond to distinctive positions in the *hyper-book*.

Assuming that:

- k denotes the number of double pages in a *hyper-book*,
- I is an ordered set of indices denoting a position in the *hyper-book*, where the elements of the set I range from 0 to the number of double pages plus 1, in order to cover all the possible positions existing in the *hyper-book* (a front cover, k double pages and a back cover), i.e.

$$I = \{0 .. k+1\}$$

the set P of positions of the *hyper-book*, containing all the possible locations of the *hyper-book*, can be defined as:

$$P = \{pos_i, \forall i \in I\}$$

The set I is referred to as the set of position indices, and characterises some relevant positions. These positions include, for instance, the front cover pos_0, the back cover pos_{k+1}, the Table of Contents pos_{toc} (where toc denotes an index, $toc \in I$, $0<toc<k+1$), etc.

Figure 4.9 summarises the concepts of positions and indices.

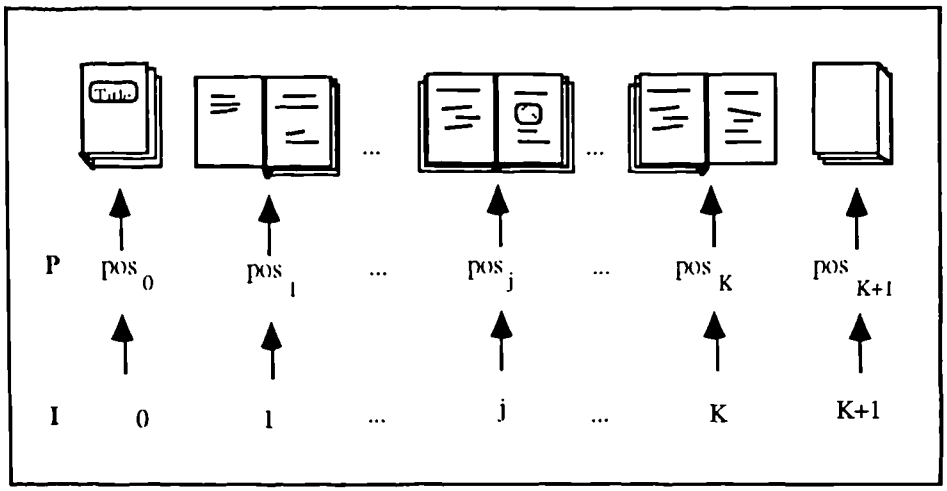


Figure 4.9: Positions in the hyper-book.

Some attributes are associated with a position and influence how it appears, i.e. they contribute to the "state of appearance" of the position itself. They include personalisations and explanations.

Personalisations deal with three different types of object (bookmarks, notes, and annotations) as described in section 4.2.2.

Explanations consist of descriptive texts which are shown in overlapped windows (explanatory windows) when an active text of type <w-text> is activated (see section 4.2.1).

Given any specific position pos_i, it is possible to define a set of personalisations $Pers_i$ associated with the position pos_i as the tuple:

$\forall i \in I$,

$Pers_i = <Bookmarks_i, Annotations_i, Notes_i>$

where

- $Bookmarks_i$ indicates the set of bookmarks defined by the user in the position pos_i;
- $Annotations_i$ indicates the set of user's annotations defined in the position pos_i;
- $Notes_i$ indicates the set of user's notes defined in the position pos_i.

In addition, analogously to the personalisation set, it is possible to define the set $Expl_i$, which indicates the set of explanatory windows opened in the position pos_i.

The Appearance State

Having defined the positions and their personalisation and explanation attributes, it is possible to introduce the concept of appearance state, which represents how information is visualised on the screen. An appearance state is defined as a particular position, with its personalisations and its explanations.

In other words, the set S of appearance states may be formalised as:

$S = \{s_i = <pos_i, Pers_i, Expl_i>, \forall_i \in I\}$

The relevant positions indicated above specify relevant appearance states. For instance, s_0 indicates the appearance state associated with the position pos_0, i.e. the front cover.

When a user interacts with the *hyper-book* system, s/he is presented with one appearance state, referred to as the current appearance state associated with the current position. It is possible to pass from one appearance state to another, by applying a number of operators. The flow of actions and transitions generated by means of these operators correspond to user interactions. In terms of the *hyper-book* model, this means that each transition of the system from a state of appearance to another state is determined by a user stimulus. The set of all the possible user stimuli, i.e. external inputs for the system, will be referred to as the set U of user inputs.

Operators

It is possible to identify different types of operator: direct access, linear access, links, history, personalisation, and searching. A formal description is given according to a usual notation for functions in the form:

operator-name: A -> B.

This will denote a transformation function from the set A into the set B.

Moreover, in the rest of this section, s_n will be used to indicate the current state of appearance, where $0 \leq n \leq k+1$.

direct-access: S x I -> S

Direct-access indicates an operator which maps the current appearance state and a position index onto another appearance state, associated with that position index.

This may involve access to a specific logical part of the *hyper-book* (such as Tables of Contents, Index, front cover, etc.) or to a physical location inside the *hyper-book* (e.g. by specifying the page number). This address may be determined by a user input. In order to be able to apply the *direct-access* operator, a logical or physical address needs first to be translated into a position index.

Assuming that $i \in I$, then

direct-access(s_n, i)=s_i

linear-access: S x D ->S

Linear-access indicates an operator which maps the current appearance state s_n, associated with the position pos_n, and a moving direction (next or previous) onto another appearance state, associated with the next position pos_{n+1} or the previous position pos_{n-1} in the ordered set I, according to the moving direction.

When the operator is applied respectively to the first/last position (front cover/back cover) and the moving direction is respectively previous/next, then the position, and therefore the appearance state, does not change.

Assuming that:

D={next, previous} is the set of moving directions which the user can select, $d \in D \subset U$, then

$$\text{linear-access}(s_n, d) = \begin{cases} s_{n+1} & \text{if d=next} \\ s_{n-1} & \text{if d=previuos} \end{cases}$$

Links

There are two types of link operator, which map the current appearance state and an active text onto another appearance state, which depends on the active text activated. In some cases the position does not change during the transition, in others the position changes. The two types are defined below.

hj-link: S x T' -> S

The *hj-link* operator maps the current appearance state s_n and an active text of type <h-text> or <j-text> (i.e. a source of a hierarchical or jump link) contained in the position pos_n, onto another appearance state, associated with a new position. The new position is defined by the link destination.

Assuming that:
T' is the set of text of type <j-text> and <h-text>,
$t \in T'$,
r: T'-> I is a function which maps an active text of type <j-text> and <h-text> onto a position index of I,
then

$\text{hj-link}(s_n,t)=s_{r(t)}$

w-link: S x T'' -> S

This type of link operator maps the current appearance state s_n and an active text of type <w-text> (i.e. a source of a window link) contained in the position pos_n, onto another appearance state, which has a different set of explanatory windows, but maintains the same position and personalisation sets. The set of explanatory windows does not change if the window associated with the activated text is already open.

Assuming that:

T'' is the set of active text of type <w-text>,

t∈ T'',

W is the set of all possible explanatory windows in the *hyper-book*,

p: T'' -> W is a function which maps an active text of type <w-text> onto an explanatory window,

p(t) = w, is an explanatory window,

then

$$\text{w-link}(<pos_n, Pers_n, Win_n>, t) = \begin{cases} <pos_n, Pers_n, Win_n> & \text{if } w \in Win_n \\ <pos_n, Pers_n, Win_n'> & \text{if } w \notin Win_n \end{cases}$$

where

$Win_n' = Win_n \cup \{w\}$ is the new set of explanatory windows obtained by adding the new window w to the previous set of explanatory windows.

history: S x H -> S

History indicates an operator which maps the current appearance state onto another appearance state, associated with a position previously visited. The ordered set H contains the indices of the previously visited positions up to the current one (pos_n).

Assuming that h∈ H, then

$history(s_n,h)=s_h$

personalisation: S x C-> S

Personalisation indicates an operator which maps the current appearance state onto a new appearance state, which varies its personalisation sets but maintains the same position and explanatory windows. The variation depends on the type of personalisation, which is determined by a user input.

Assuming that:

C={bookmark, note, annot} is a set which indicates the possible types of personalisations,

c∈ C,

s_n = <pos_n, $Pers_n$, Win_n> represents the current appearance state, where pos_n is the current position, $Pers_n$ is the set of personalisations associated with the position pos_n, Win_n is the set of explanatory windows associated with the position pos_n,

then

personalisation(<pos_n, $Pers_n$, Win_n>, c) = <pos_n, $Pers_n'$, Win_n>

where

$Pers_n'$ is the new set of personalisations obtained by adding or removing an element in the set of personalisations, according to the type c (e.g. bookmark, note).

searching: S x Q -> S

Searching indicates an operator which maps the current appearance state s_n onto the new appearance state which matches the query, which is provided as a user input. A query consists of a search term or a Boolean combination of search terms.

Assuming that:

Q is the set of possible queries,

$q \in Q$,

se: Q x I -> I is a function which maps a query and the current position index onto a new position index, associated with the query $q \in Q$, where the search term appears,

then

searching(s_n, q)=$s_{se(q,n)}$

For the sake of completeness, other operators should be included which give access to external sources (e.g. printer, word processor, library). This would involve the extension of the *hyper-book* model to include other elements of the *external world*. In order to keep the model simple, this kind of external sources operators have not been formally described.

4.2.3.3 Formalising the Functional Behaviour of the Hyper-book

Having defined both the structural and functional components of the *hyper-book*, this section completes its description as a dynamic system by providing a formalisation of its functional behaviour. The functional behaviour of a system deals with its use and is concerned with how the system evolves. In order to describe this process it is necessary to define other sets and functions.

As the *hyper-book* is a direct manipulation system, users directly interact with the system entities and operators, which are graphically represented, by means of buttons. The activation of a button leads to a user's desired action being accomplished. This accomplishment is carried out by the application of one of the operators described in the previous section.

Let OP be the set of all the operators defined above, i.e.

OP={direct-access, linear-access, hj-link, w-link, history, personalisation, searching}.

At this stage it is necessary to introduce a new function, which selects and activates an operator on the basis of a user input.

Let us define the command function *Com*:

Com: U -> OP

where U is the set of user inputs, and OP is the set of operators.

This function transforms a user input into the activation (i.e. execution) of an operator op∈ OP. By definition, any possible operator selected in OP will operate on the current state s_n and on another set $\underline{X}$, which may be one of the sets I, D, T', T'', H, C, or Q, as specified in the definition of the operators.

Assuming that u∈ U is a user input , and op∈ OP then

Com(u) = op

Having selected and hence activated an operator, the system changes state, passing from a current state of appearance s_n into a new state of appearance s_n'. This new

state of appearance needs to be made visible to the user by showing it on a display. In order to formalise this operation, the visualisation function *Vis* is introduced:

Vis: S -> O

where S is the set of appearance states, and O is the set of outputs of the system which are connected to a display, where the *hyper-book* graphical interface is presented.

The *Vis* function shows the new state of appearance of the *hyper-book* on the display to the user. The new state of appearance, once visualised, becomes the new current appearance state.

The rest of this section analyses how the *hyper-book* system evolves from an initial state towards other states under certain inputs, which correspond to the user interactions.

The *hyper-book* system starts from the state which presents the front cover of the book. According to the previous formalisation, it is referred to as the initial state of appearance s_0.

A user starts his/her interaction with the *hyper-book* system by providing it with an input. This input represents a desired action, such as turning a page, add an annotation, following a link, etc. As mentioned above, each time the user provides an input to the system, s/he activates the execution of an operator through the *Com* function.

The selected operator, op$\in$ OP, will be applied to the current state of appearance s_n and to another set $\underline{X}$, according to the specific operator selected. As mentioned above, the set $\underline{X}$ may be one of the sets I, D, T', T'', H, C, or Q.

On the basis of the previous definitions, the *hyper-book* behaviour can be summarised as the following three-step process: a user input command, a transformation of the current state of appearance by applying an operator, and the visualisation of a new appearance state on the display in the form of an output image o$\in$ O.

Assuming that:
$u \in U$, $op \in OP$, $n' \in I$, $0 \leq n' \leq k+1$, $o \in O$, then

1- *Com* (u) = op	Input
2 - op (s_n, $\underline{X}$) = $s_{n'}$	Transformation
3 - *Vis*($s_{n'}$) = o	Output

The output may generate a new input from the user. The cycle can then be repeated, changing iteratively the current appearance state.

Figure 4.10 graphically represents the *hyper-book* system described so far, indicating the relations between its components. For reasons of simplicity, the *hyper-book* state represents not only the appearance state, but also the sets H, T', etc. mentioned above, although they are not used by the *Vis* function.

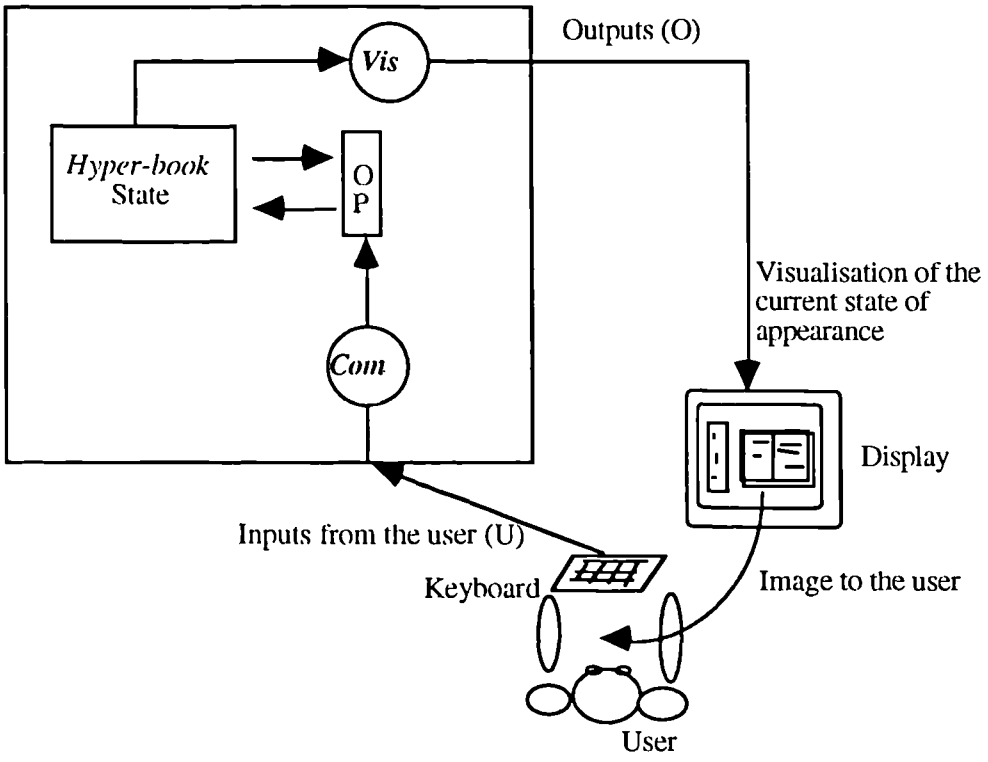


Figure 4.10: The hyper-book system

Finally, on the basis of the above descriptions, it is now possible to define the *hyper-book* system as the tuple:

hyper-book = <Hyper-book-Structures, OP, U, O, *Com*, *Vis* >,

where Hyper-book-Structures denotes the structural components of the *hyper-book* described in section 4.2.3.1, OP is the set of operators, U is the set of user inputs, O is the set of outputs of the system,*Com* is a function which selects and activates an operator on the basis of a user input, and *Vis* is a function which shows an appearance state on the display, as described above.

4.3 Hyper-books in the Context of the Electronic Library

Hyper-books are conceived as entities of an organised collection, an electronic library, which inherits most of the features of the physical library but introduces a number of new properties which result from its non-physical nature.

A number of advantages are related to the potential of *hyper-books* in the library both when a new book becomes part of the library and when the book is actually used by a user.

The acquisition phase is facilitated by having a tool such as the *hyper-book builder*, which incorporates all the functions necessary for converting existing electronic material into a format suitable for the library. Information provided in the document's original markup could also be exploited for automatic cataloguing, indexing (e.g. CLARIT (Evans et al. 1991) and SIMPR (Gibb and Smart 1991, Gibb 1993) projects) and abstracting. The CIP element in the markup document is very useful for the cataloguing process as it allows the bibliographic description provided by elements in the title and cover page to be enriched with other information, such as the classification code.

A number of other advantages are related to the potential of *hyper-books* in the library when they are actually used:

• *hyper-books* do not suffer from the constraints of the physical books and can be replicated as many times as required;

• the library could be accessed remotely from any location through a network using personal or portable computers. Universal accessibility represents the most revolutionary aspect of the electronic library;

• *hyper-books* are not isolated objects and links can exist between documents. The references of a book could act as a citation index; a bibliographical item represents the source of a link to the referred document. In this way a network of interconnected documents may be created. An important issue is the need for uniquely identifying a book in the library. Different alternatives are possible. ISBN is a unique code for books, but not all books (particularly older ones) may have it. Another possibility is to define an algorithm for automatically computing unique codes starting from information about a book (e.g. title, author, publisher, date). In-house accession numbers could be also employed, but problems could arise when several libraries have to be interconnected;

• the concept of loan in an electronic environment is different from the traditional concept. Loan is comparable to having temporal access to the electronic library material. As documents can exist in as many copies as requested, books are always available. When a book is on loan, personalisation tools are available to the reader. Reader annotations do not affect the original copy, but they are saved as user personal comments. Personalisation tools could have several potential developments in an electronic library: for instance, a tutor could annotate a set of books and make them public to a group of students.

The concept of buying a book in an electronic environment could be very similar to borrowing a book; however, the book, in this case, is definitely provided to the reader. The book is physically copied in a user personal library, where it can be manipulated through annotation, cut and paste operations, etc. A charging system may need to be defined in order to manage the purchase of books. Once the book has been bought the user becomes its owner;

• users are understood to have different interests and objectives. By observing user behaviour, the library will be able to construct individual user profiles. Knowledge about user needs may be used in order to optimise retrieval strategies and acquisition policy. Such reader profiles can be used to notify new publications of interest to the user and to rank documents, identified during search, in order of relevance. Reader profiles can also be used to filter library material and reduce the confusion caused by an excess of information, or to create contacts among users sharing the same interests;

• several strategies may be available in an electronic library in order to access the library material. Some of them are inherited from the physical library, while others are

only available in the electronic environment. In a physical library, there are different access strategies: for instance searching through the catalogue, asking the librarian, and browsing through the library shelves. In the context of an electronic library, catalogue searching is already widely used. The librarian's role as a consultant (reference librarian) has been recognised to be very important especially with large collections: librarians have some knowledge of the information resources of the library and a responsibility to facilitate their use by clients. Intelligent tools (expert systems) are being developed in order to simulate the human reference librarian in electronic libraries (Morris 1992): they interact with users in order to understand the user query, they help in understanding the result of a search, they provide general information about the system, etc. Browsing through the library shelves is another important method for finding useful material in a physical library. The book dimension, the title, the thickness, the physical attributes of the cover page (colours, visual appearance, material, etc.) represent important visual clues for recognising a book in a collection. The provision of a similar access method in an electronic environment involves the design of a virtual library, where rooms, shelves and books are reproduced by simulating a three-dimensional environment (the potential of a virtual library was investigated in the Super-Library project).

Other strategies for accessing the library material, which are not available in the physical library, include: activating a link to an external document from a book which is currently open, and using the facilities already mentioned about user profiles (i.e. having a filtered access to the library material, being notified of new publications, etc.).

4.4 Analysis of How the Hyper-book Matches the Book Metaphor

As Hammond and Allinson point out (Hammond and Allinson 1987), there are two relevant dimensions for understanding the information which metaphors convey: scope (i.e. the number of concepts addressed) and level of description (i.e. the type of knowledge conveyed) as indicated in section 2.3.1 of chapter 2.

They propose four different levels: task, semantic, lexical and physical. In order to better describe how the book metaphor has been interpreted in *hyper-books*, an analysis is proposed here of how the *hyper-book* matches the book metaphor from these four levels of description.

Referring back to the definitions presented in chapter 2,

- the *task* level concerns the structure and constraints which the system imposes on potential user tasks. This level identifies the set of tasks the user can carry out using the system.

- the *semantic* level defines the facilities provided by the system in order to accomplish the tasks mentioned above. This level clarifies the meaning of the operators which are defined in the system.

- the *lexical* level defines the terminology or language used for referring to the system entities and operators. This level has many relations with the implementation. Terminology could refer to command names, if the interaction method is based on command languages, or to icons, if direct manipulation represents the interaction strategy.

- the *physical* level specifies the physical representation of the interface and the actions performed by the user. This level is not considered in this context because it is strictly related to the implementation and will be described in the next chapter.

4.4.1 Task Level

The tasks which have been considered in the *hyper-book* model are the typical tasks of a user who takes a book from the library. Two main approaches are foreseen: retrieving a book never seen before and retrieving an already known book. In the first case typical tasks include browsing for finding out whether the book could be interesting and which kind of topics are discussed. In the second case, typical tasks could be searching and analysis. Most of the tasks are provided in both the system (*hyper-book*) and the metaphor domain (paper book). However, the electronic medium gives the possibility of directly navigating in the library through connected books, saving time associated with retrieving related publications in a paper library. In addition, the powerful facilities (listed in the next section) of the *hyper-book* facilitate searching and navigation in the *hyper-book* itself.

The underline style has been used in table 4.1 whenever there is a match between the system and the metaphor domain.

	HYPER-BOOK	PAPER BOOK
browsing	find out the main topics covered by a book find out if the book discusses a specific topic direct access to related publications direct access to any publications referred to in the book	find out the main topics covered by a book find out if the book discusses a specific topic
searching	retrieve information contained in the book wherever such information occurs in the book	retrieve information contained in the book by looking through the Table of Contents or the Index
analysis	study the book sequential and non-linear reading highlight important parts in the book exporting portions of text for successive elaboration	study the book mainly sequential reading highlight important parts in the book

Table 4.1: The task level

4.4.2 Semantic Level

The semantic level identifies the set of facilities the system provides in order to accomplish the tasks mentioned above. The distinction between the task and semantic level may occasionally be difficult to make, because there could be a direct correspondence between a task and a facility which accomplishes it. The underline style has been used in the table whenever there is a match between the system and the metaphor domain. The following tables indicate clearly that most of them do not have a correspondence with the paper book: print, search, navigation, history and link. These should allow most of the tasks (available in both the domains) to be accomplished faster and more easily. For instance, the search facility in *hyper-books* facilitates retrieval of information when the Table of Contents and Index are not sufficient. Other facilities allow other tasks to be available only in the *hyper-book* domain (e.g. direct access to related publications). The facilities mentioned in the following have been already described in terms of reader services in section 4.2.2.

	HYPER-BOOK	PAPER BOOK
Navigation	direct access to a special location (Table of Contents, List of Figures, List of Tables, Index)	access to a special location (Table of Contents, List of Figures, List of Tables, Index)
	direct access to a page with a given page number	access to a page with a given page number
	open the book at random (active thickness)	open the book at random
	go to the previous / next page	go to the previous / next page
	flip pages	flip pages
	direct access to objects referred to in the text	access to objects referred to in the text
	zoom figures and tables	
	direct access to a section, a figure or a table from the Table of Contents, the List of Figures or the List of Tables	access to a section, a figure or a table, having checked the page number in the Tables of Contents, List of Figures or List of Tables
	direct access to a reference from the index	access to a reference from the index
	close and open the book	close and open the book
	direct access to related publications in the library	
Orientation	consult the Table of Contents	consult the Table of Contents
	check the current position through header, page number and thickness	check the current position through header, page number and thickness
History	go to the last visited page	
	go to a page chosen among the last visited page(s)	
	open the book where a bookmark has previously been inserted	open the book where a bookmark has previously been inserted
Personalisation	insert/take out a bookmark	insert/take out a bookmark
	annotate the book	annotate the book
	add/delete overlapped notes	stick/remove a paper note

Searching	full text search	
External sources	copy text automatically access to a word processor print	copy text manually

Table 4.2: The semantic level

4.4.3 Lexical Level

This level identifies the language used to refer to the system entities and operators. The interaction strategy used for *hyper-book* is based on direct manipulation. This means that users directly interact with the system entities and operators which are graphically represented.

A syntactic perspective, which would indicate whether the language is correctly used (e.g. whether lexical elements are in the right sequence) has not been considered in this context because most of the user operations (e.g., go to the back cover, flip pages) are subsumed as a unit in the action of activating an icon. Most of the user actions are intrinsically syntactically correct, because a sequence of syntactically correct procedures are associated with the icon (Hammond and Allinson 1987). The syntactic level can be viewed as being trivial or surface-near in the sense that the user has only to understand the correct sequence of a point and click action or the requirement to input information in a dialogue box in order to invoke a system action. A syntactic perspective could be introduced when considering an interaction strategy based on command languages, where the user has to know exactly the syntax of commands.

The lexical level in *hyper-books* mainly identifies buttons and graphical objects with which the user can interact. A book image is presented on the screen. All the structural entities which are parts of the *hyper-book* (pages, tables, figures, text, front cover, etc.) are graphically represented as in a paper book using an analogous lexicon. They could not be inserted in a table for practical reasons but they will be fully visualised and described in the following chapter.

With regards to the operators, although they have a semantic equivalent in the paper book, they do not have a correspondence at the lexical level. The lack of an explicit mapping results from the transition from a real world to a virtual world, which

requires the introduction of new objects (operators) which subsume real actions in the physical world. Therefore, in the following tables, the paper book column is left empty.

The operators which act on the system entities are usually indicated by a tuple: <icon, name> (when they are outside the book space), or simply by an icon or by a graphical object (when they are inside the book space). The table below contains all the icons and names of operators which are outside the book space, i.e. in the reader tool area. Furthermore, it contains the icons of operators which are embedded in the book representation. These operators do not have a name; a brief description is in square brackets to indicate that it is not part of the lexicon. Even in this case, the classification presented in section 4.2.2 has been mostly maintained. However, navigation tools have been further subdivided into several subclasses: direct internal access, direct external access, linear access, transverse and hierarchical links. This more refined classification has been used during the evaluation phase to measure the frequency of tool usage.

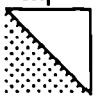
HYPER-BOOK	PAPER BOOK
 flip	
 [page ears on the left-bottom corner of the book for indicating the previous page operator]	
 [page ears on the right -bottom corner of the book for indicating the next page operator]	

Table 4.3: Navigation: Linear Access

HYPER-BOOK	PAPER BOOK
 front cover  back cover  go to page...  Index  List of Tables  List of Figures  Contents  [thickness]	

Table 4.4: Navigation: Direct Internal Access

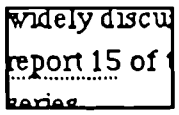
HYPER-BOOK	PAPER BOOK
 Bookshelf  [special text style (dotted underline) for indicating link sources]	

Table 4.5: Navigation: Direct External Access

HYPER-BOOK	PAPER BOOK
in figure 3, th book general [special text style (dotted underline) for indicating link sources] 1.Introduction 2. chapter 1: the I 2.1 The design [each entry in the Table of Contents, List of Tables, etc.] [active figures and tables]	

Table 4.6: Navigation: links

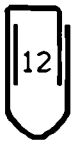
HYPER-BOOK	PAPER BOOK
 return  recent  [bookmark]	

Table 4.7: History

HYPER-BOOK	PAPER BOOK
 Search	

Table 4.8: Searching

HYPER-BOOK	PAPER BOOK
 Add Notes  Delete Notes  Add bookmark  Delete bookmark  Annotations	

Table 4.9: Personalisation

HYPER-BOOK	PAPER BOOK
 Copy Text  Print pages  exit  Word 5.0	

Table 4.10: External Sources

The icons and terms presented so far at the lexical level anticipate implementation details of the *hyper-book* system better described in the next chapter.

As mentioned earlier, the *physical* level, which is completely dependent on implementation choices, in terms of interface design and interaction strategies, has not been considered here and will be described at the end of the next chapter.

Chapter 5:
Hyper-book Implementation

This chapter describes the prototype which has been developed for the purpose of evaluating the *hyper-book* model. The system has been implemented on a Macintosh IIx; the screen size is 19 inches and the resolution is 72 dpi. The software platform is HyperCard 2.1, which has was chosen because it allows a working system to be developed in a short time by providing an easy to use programming language (HyperTalk) and various built-in tools for graphical interface design. Another software package, the XGML Translator from the Exoterica Corporation, has been used for parsing and specifying conversion rules of SGML marked documents.

Each *hyper-book* instance[1], including the empty template, consists of separate files (HyperCard stack). Figures and tables are contained in separate PICT files and made visible in the stack by means of external commands[2].

The user interaction style is based on direct manipulation, which provides an immediate and natural understanding of the system functionalities compared, for instance, to command languages, menu selection or other less natural techniques.

---

[1] *Hyper-book* is the general model defined for representing electronic books; a *hyper-book* instance represents a particular electronic book which has been implemented according to the *hyper-book* model.

[2] External Commands are compiled programs written in C or Pascal, which are developed as an extension of HyperTalk.

The organisation of this chapter follows the structure of chapter 4, but presents the same topics from the implementation point of view: it contains two main sections which describes the *hyper-book* building process and the *hyper-book* browsing process.

5.1 Hyper-book Builder

5.1.1 The Empty Template

As described in the previous chapter, an empty template is an empty container designed specifically to hold the book. For the implementation phase, an empty template was defined which consists of reader services, editor tools and three principal page models.

Both the reader and editor tools are provided by means of active buttons whose name and icon clarify their purpose. Some scripts are linked to the buttons themselves and to other objects which are in the stacks (cards, fields, etc.). Most of the icons were taken from a set of predefined icons in the HyperCard environment; other icons were designed ad hoc following the same style.

The page models include:

- the title page, which contains the following components (represented by HyperCard fields) defined in the *hyper-book DTD*: the title, the sub-title, the author(s), the editor(s), the publisher, the place of publication, the date of publication, a component for additional information, etc.;

- the verso page, which contains the following components defined in the *hyper-book DTD*: publisher details, printer details, copyright details, the ISSN, a component for additional information, etc.;

- the text page which consists of the footer, the header, the text space (see figure 5.1).

Each of the elements which may occur in the page models mentioned above has some predefined features: size, a position inside the page and some typographical information related to the text they contain: the typeface, style, size, line height, etc.

In addition, the empty template contains a number of hidden containers, which are used for storing information about links (*link-info container*), section headers and their identifiers (*header-info container*), figures and tables (*picture-info container*) and a temporary container for the text used during the pagination process.

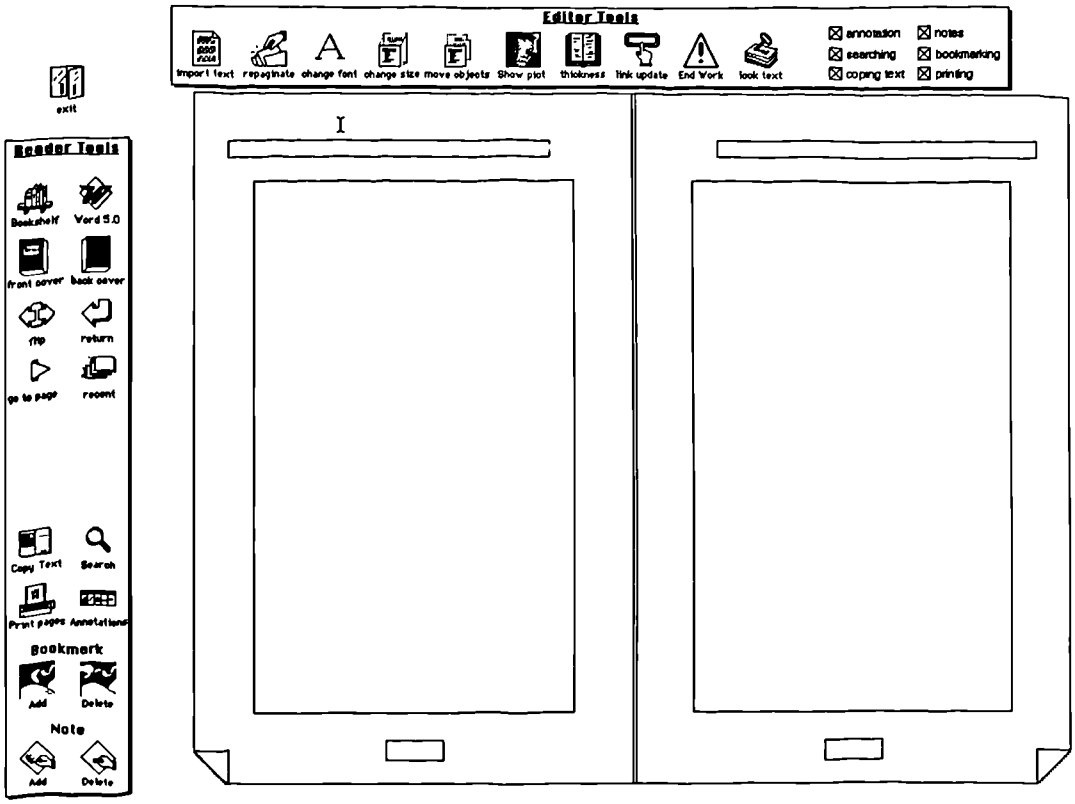


Figure 5.1: The empty template

5.1.2 Hyper-book Generation Process

This section describes how the different steps of the *hyper-book* generation process are implemented. As indicated in figure 4.8 of the previous chapter, in the first step (automatic translation) an SGML document is parsed, to verify that it is conformant with the *hyper-book DTD*, and then translated into the *hyper-book* format according to a set of rules defined using the Exoterica XGML Translator, which allows actions to be specified which are executed when an event occurs in an SGML document. A frequently used event is an ELEMENT event. A rule in this case will specify which actions have to be executed when a particular element occurs. The set of these rules constitutes an XGML program.

At the implementation level, the *automatic translation* step defined in the design is composed of two separate processes: the first one takes as input the *hyper-book* DTD, a document marked according to such a DTD and the XGML program. Both the DTD and the XGML program are compiled, the document is parsed and then translated. The result of this translation are intermediate files. These intermediate files, together with the empty template represent the input for the second process which generates the

actual *hyper-book*, thus completing the automatic translation process. The automatic translation process is activated through a stack, Book-Builder, which contains two buttons for executing the two processes mentioned above.

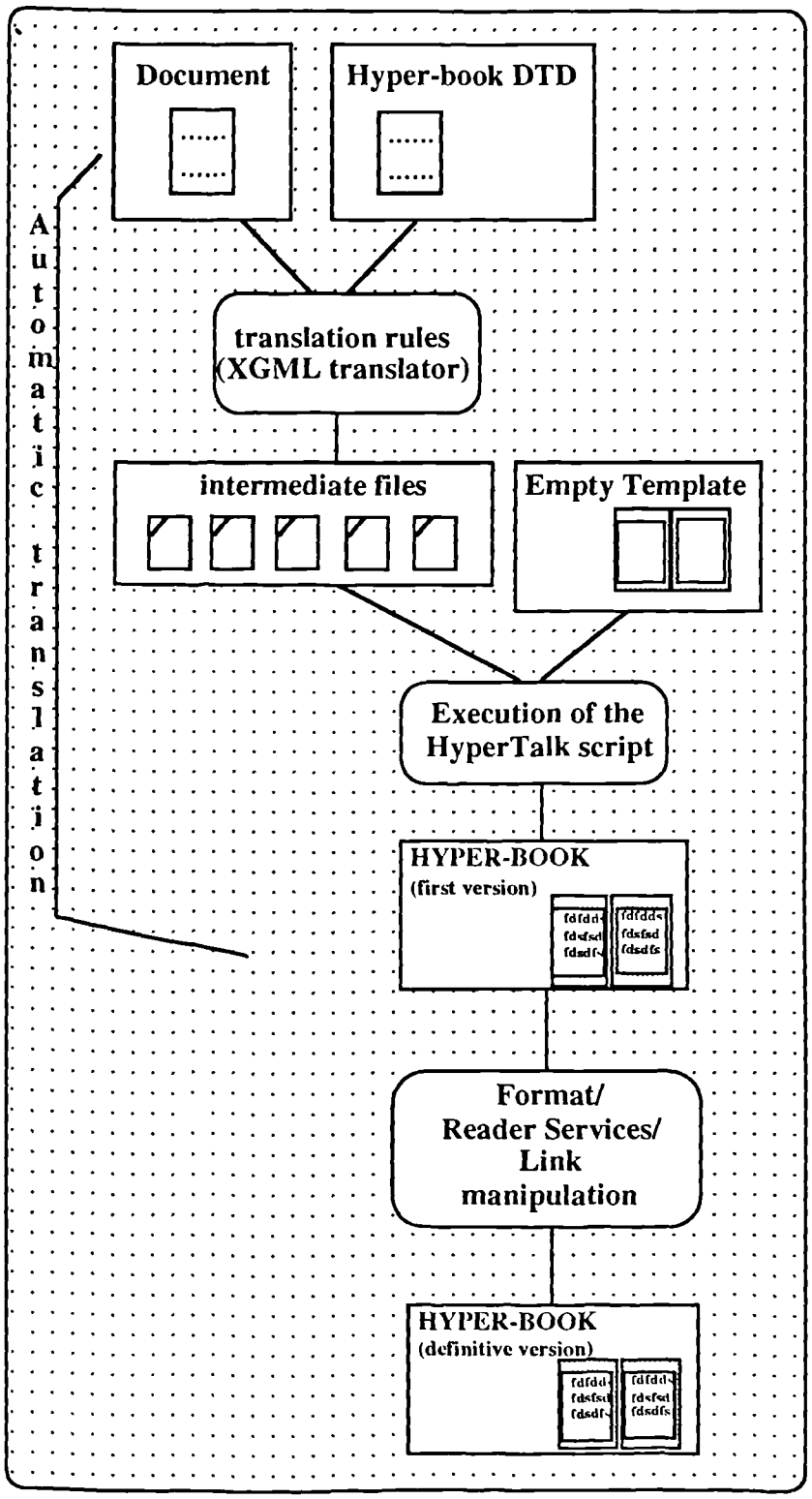


Figure 5.2: Hyper-book generation process

The intermediate files are:

- a set of text files: foreword, preface, abstract, main body, appendix, references, glossary, footnotes, etc. which are extracted from elements of the original SGML document.
- a hyper-talk script, i.e. a set of instructions which, when executed, allow the *hyper-book* to be created by importing the text files mentioned above.

The script contains some *simple* instructions for inserting text in predefined containers (HyperCard fields) followed by a number of more *complex* instructions which usually expand the stack size by inserting new cards. Referring to the structure of the empty template, the title and the verso page are filled using simple instructions such as *put [author's name] in field authors of card title page*. The same kind of instruction is also used for storing information about links, section headers and their identifiers, figures, index entries and acronyms in the special hidden containers (e.g. *link-info container*). Figures, tables, sections and subsections have an identifier in the original marked up text which is unique but sometimes is not meaningful to the reader. Therefore a new identifier is linked to sections and subsections which takes into account the section order and level (e.g. 1., 1.2, 4.1.3, etc.). Figures and tables are simply numbered by taking into account the order of occurrence. A map is maintained between the old and the new identifiers.

The *complex* instructions are essentially the *import* and the *createspace* commands. The *createspace* command is used with empty elements (Table of Contents, List of Figures, Index) for creating an empty page with a title which will be filled later with an automatically generated text. Such a page is a sort of marker which indicates where an empty element should occur as defined in the original SGML document. The *import* command is used for importing a text into the *hyper-book*. This command is invoked for each of the text files which have been generated as intermediate files and behaves differently according to the specific file, because each file requires a different treatment. That is why the text of the document has been fragmented into several files during the first process of the automatic translation.

For instance, when the main body or the appendix are imported, the new identifiers of (sub)sections are inserted before the (sub)section headers and the new identifiers of tables or figures appear below a table or a figure (e.g. Table 3, Figure 6, etc.). A bold style and a bigger size is associated with the (sub)sections headers. Link sources are

given a special style (group) in order to be selectable as a unit and to be recognisable in the text.

Acronyms are also automatically identified in the text. When the acronym file is imported, a bold style is attributed to the acronym terms in order to emphasise them.

In general, when a new file is imported, the text is firstly put in a hidden scrolling field, where the changes mentioned above are applied and then it is paginated. The *import* command can also be used alone for importing any kind of ASCII text. In this case no structural information is included; however the text is paginated and the reader and editor tools are still made available.

An example of a typical script is given below, the first four lines are simple instructions, the other are complex instructions:

```
put "Hyper-Books in an Electronic Library" into field TITLE of card TITLE PAGE
put "Nadia Catenazzi " into field authors of card TITLE PAGE
.....
put "02484-8015" into field ISSN of card VERSO
.....
put "xi,•(see chapter 1),U1" after field LinkInfo of cd TEMPOR
..
import(ABSTRACT)
import(PREFACE)
createspace(CONTENTS)
import(TESTO)
import(REFERENCES)
...
```

When all the files have been imported, further commands are invoked in order to generate the headers and footers (page numbers). In addition, the Table of Contents, the Index, the List of Figures and the List of Tables are automatically produced and the corresponding activation buttons are added to the reader tool area. The Table of Contents is generated starting from the headings of the different sections and subsections defined in the book. The Index is generated starting from the information extracted from the marked text and stored in a special hidden container, the index terms are sorted and linked with the corresponding parts in the text.

Obviously, with some elements being optional as defined in the *hyper-book DTD*, their generation is only activated if they were present in the original SGML document.

The automatic translation step is therefore managed partially using an XGML translator, and partially using HyperTalk scripts. This is due to the fact that the XGML translator allows SGML files to be transformed into a file or a number of files with a format specified by the translation rules. However in order to automatically

generate a *hyper-book*, it is necessary to execute a HyperTalk script which allows a stack (i.e. the empty template) to be changed dynamically by importing text.

The other steps are the same as described in the design chapter. The manipulation of the *hyper-book* format, the reader services and the link structure are based on the editor tools, which will be described in a following section. The indexing process is not mentioned in the diagram. HyperCard automatically indexes all text elements which are in a stack and incorporates a built-in searching facility. The current prototype exploits such facilities. Figure 5.2 illustrates the *hyper-book* generation process.

5.1.3 Link Definition

The different types of link are indicated in different ways in the original marked up document (see section 4.1.4 and 4.2.1 of the previous chapter, and the *hyper-book DTD* in Appendix A). *Bibliographical link* and *note link* are empty elements, i.e. elements without contents, and have an IDREF attribute which identifies the destination (bibliographical item or footnote).

When a *bibliographical link* is imported into the *hyper-book*, a link source is inserted to replace the empty element. The link source consists of the bibliographical identifier in square brackets. Information about the link is added to the *link-info container* in terms of link type, *b*, and link source (e.g. [WALKE-91] in figure 5.3). The link destination is omitted as it may be found by searching, in the references, for the bibliographical item which starts with the same identifier used for the link source. Therefore the bibliographical identifier is the only information needed to locate the link destination.

When a *note link* is imported into the *hyper-book*, a link source is inserted to replace the empty element in the marked document. The link source consists of the footnote identifier subscripted and in brackets. Information about the link is added to the *link-info container* in terms of link type, *n*, and link source. The link destination is omitted as it may be found by searching, in the footnote list, for the item (footnote) which starts with the same identifier used for the link source.

An *internal text link* corresponds to an element in the markup document which contains the source of a link to a structural element of the document, such as a section.

It has an attribute, IDREF, which identifies the destination of the link. When an *internal link* is imported into the *hyper-book*, information about the link is added to the *link-info container*, in terms of link type, *xi*, link source and link destination.

A *graphic link* corresponds to an element in the markup document which contains the source of a link to a graphic object, such as a figure. It has an attribute, IDREF, which identifies the destination of the link. When a graphic link is imported into the *hyper-book*, information about the link is added to the *link-info container*, in terms of link type, *g*, link source (e.g. •(Figure 4) as shown in figure 5.3) and link destination (e.g. *Figure 4*).

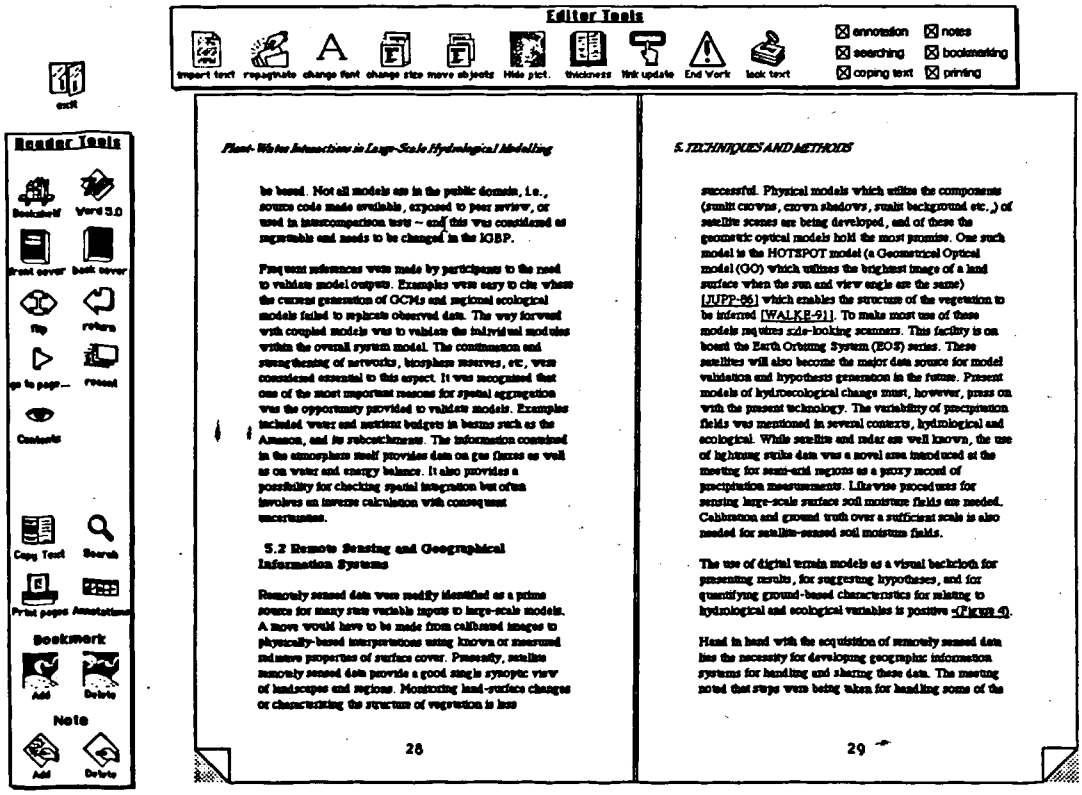


Figure 5.3: Bibliographical and graphic links

An *external link* corresponds to an element in the markup document which contains the source of a link to an external document. It has two attributes, which identify the destination of the link, i.e. the external document and, optionally, a structural element of the external document. When an external link is imported into the *hyper-book*, information about the link is added to the *link-info container*, in terms of link type, *xe*, link source (e.g. NO.7 in figure 5.4) and link destination (e.g. *Rep#7).*

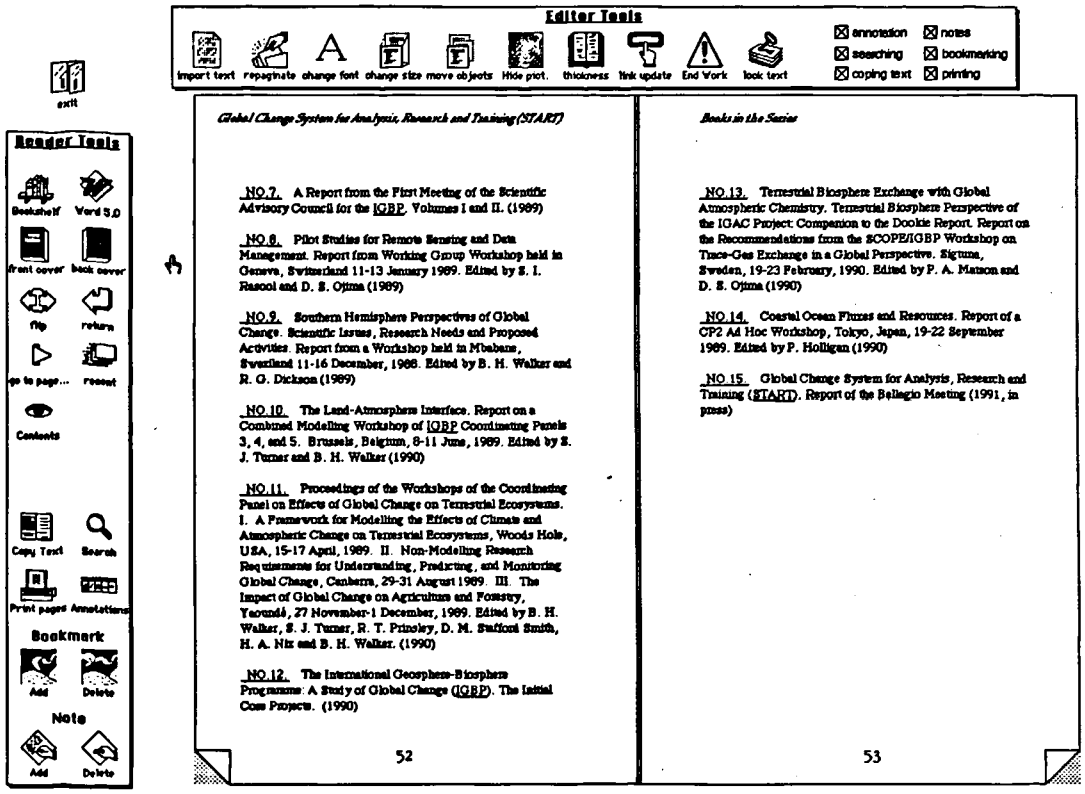


Figure 5.4: External links and acronyms

A *document link* corresponds to an element in the markup document which contains the source of a link to an item of the list of related documents. An attribute identifies the destination of the link. Information about the link is added to the *link-info* container in terms of link type, *d*, link source, and link destination.

Links to acronyms are not explicitly indicated in the marked up document. They are automatically created when a string is found in the text which matches an acronym term. IGBP in figure 5.4 is an example of an acronym.

5.1.4 Editor Tools

This section describes the editor tools which were implemented in the prototype. Most of them are the same as presented in the previous chapter, others have been added for practical reasons as clarified below.

The *import text* button activates the text importing process into the empty template, completing, in this way, the automatic translation process described in

section 5.1.2. Its script incorporates the code automatically produced during the first step of the *hyper-book* translation process. This button may be used only once, where the template is empty.


The *change font* button allows the typeface, style and size of any selected portion of text in a page (for instance the header) to be changed. If the change requires a repagination, the user is given the choice to activate the process immediately or to repaginate when it has completed all the changes, using the repaginate button. The repagination is not always automatically activated for reasons of efficiency. If the selection involves the entire text of an element, the change will affect each page which contains such an element. This tool is intended to provide some basic functionalities for handling text typeface, size and style. At the moment, HyperCard capabilities for formatting and managing text are quite limited, but can be expected to improve in future versions.


The *change size* button is used to change the size of the book, page and page components. They can be modified by clicking the change size button and dragging in the target object until it reaches the required size. If the change requires a repagination, the user is given the choice to immediately activate the process or to repaginate when it has finished all the changes, using the repaginate button.


The *move objects* button is used to change the position of the book on the screen and the position of book components inside the book. Their positions can be modified by clicking the move objects button and dragging the target object until it reaches the required position. For instance, figure 5.5 results from the application of both change size and move objects buttons to figure 5.3.


The *repaginate* button allows the text to be repaginated manually when the process has not been activated in one of the situations mentioned above (change size or change font). This tool was not foreseen in the design phase, and has been added here for reasons of efficiency .

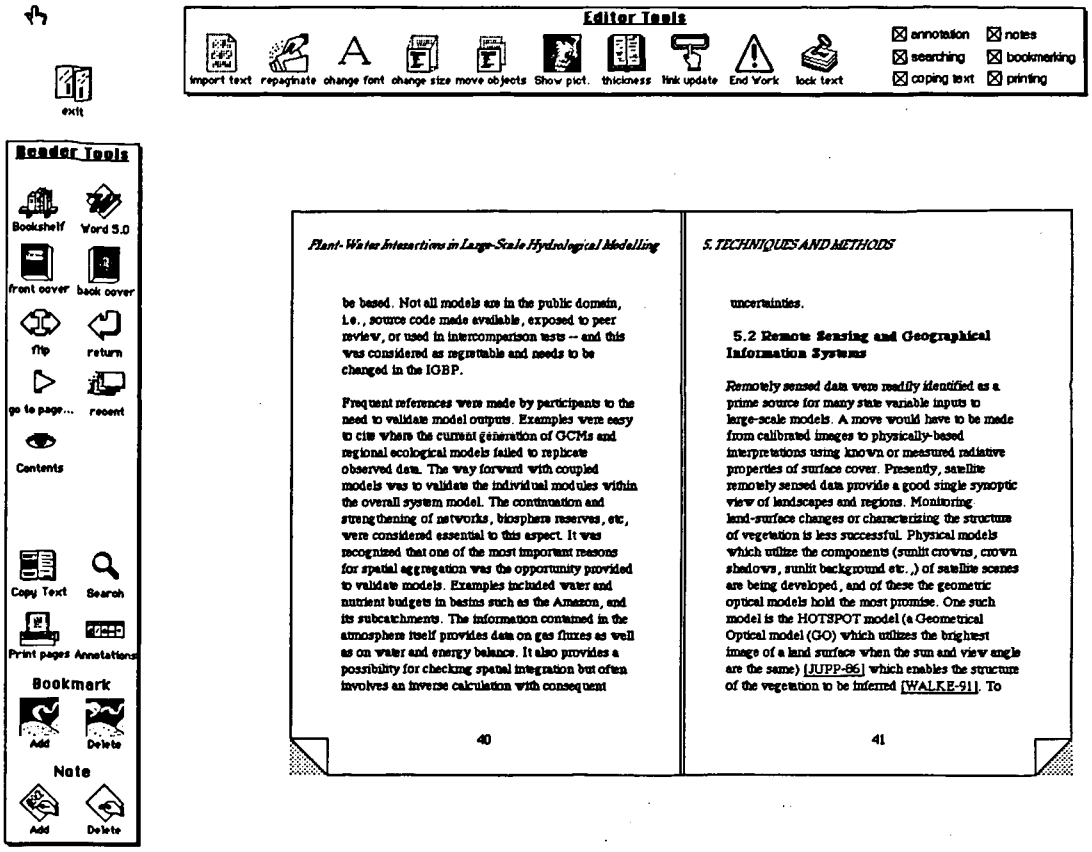


Figure 5.5: Resize and change position

The *link update* button allows the extension of the existing link structure. A new link can be inserted in the following way: first a portion of text is selected by the user; a check is made that no other links are defined over that portion; then the user is asked to enter the type and according to the type a different kind of destination has to be selected from a menu of possible destinations. For instance when a link to a table has to be defined, the list of the available tables is shown in a menu, and an internal link destination is selectable from the List of Tables entries. An external link destination is defined by selecting from a list of other documents in the library. The destination of links to bibliographical items or notes is selected from the menu of all possible bibliographical items or notes respectively and the corresponding identifier is added to the text as a source after the selection. The information about the new link is added to the *link-info container* as in the defined scheme (source, type, destination), and the new link sources are made active and visible by applying the group style to the source.

Although the new links may be of any type (graphic, note, bibliographical, etc.), it is expected that the most useful types of link are external links, which allow a book to be connected with other related publications in the library.

When a *hyper-book* is first generated, figures, tables and thickness are not visible. The following button (*show/hide pictures* and *show/hide thickness*) have been added to allow the completion of the *hyper-book* with these missing components.

The  *show pict and* the  *hide pict* buttons allow pictures and tables to made visible if hidden and to be hidden if visible, respectively; this is useful because their presence significantly slows down any operation on the book during the building phase. It is convenient to keep pictures and tables hidden and make them visible just before the book is made available to the reader. However, even if figures and tables are not visible, there is always a suitable space for inserting them when required (see pagination process in chapter 4). In order to show a figure or a table, the table or picture identifier is used as the key for finding information in the *picture-info container*. Such information identifies the PICT file where the picture is contained and gives details about the picture size. Figures and tables are both treated as PICT objects. These buttons appear alternatively in the same place. When the show pict button is activated, pictures are made visible and then the button is replaced by the hide pict button.

The  *thickness* and the  *hide thick* buttons allow respectively the thickness, together with the front and back cover of the book, to be created or deleted. Thickness represents an important orientation clue. However, if it is available during the *hyper-book* generation process, it needs to be redrawn everytime the size of the book is modified or the number of pages changes. For this reason it is suggested that it should be added with the pictures and tables before the book is made available to the reader. These buttons appear alternatively in the same place. When the thickness button is activated, the book thickness is made visible and then the button is replaced by the hide thick button.

The  *lock text* and the  *unlock text* buttons are respectively used for locking and unlocking text. The text needs to be locked in order to make link sources

active; a link is activated by clicking (double click) on it. If the text is unlocked, clicking has a different effect: it selects a word for successive editing such as for changing the style. The link activation is the only operation which requires a locked text. These buttons appear alternatively in the same place. When the lock text button is activated, the text is locked and then the button is replaced by the unlock text button.

A set of general services can be provided to the reader (notes, annotation, searching, etc.). These services can be removed by the book publisher if they are not useful for the kind of readers and application. A number of "check boxes" (radio buttons) in the editor tool area (see for instance figure 5.1), which are associated with various *reader services,* allow the publisher to select useful services or remove unrequired services for the user. An automatic mechanism for adding new services is envisaged but not currently considered. A reader service is represented as a button; in order to add a new service, an icon, a name and a script must be generated and integrated into the system.


The *end work* button: this button should be activated when the book has been completed. It is an irreversible process. All the editor tools are removed, i.e. buttons are physically deleted together with their associated code (HyperTalk scripts). The editor tools area is replaced by a library browser, when the book becomes part of a collection.

5.2 Hyper-book

5.2.1 Book Representation and Content

A *hyper-book* can be presented as closed with the front (see figure 5.6) or the back cover visible, or open with the two pages in view. There are several ways of opening the book: by clicking the *next* button (from the front cover), the *previous* button (from the back cover), the thickness, or by using the reader navigation tools described in the next section. Each element appears in the book according to the markup of the original document. Furthermore, the order of presentation of some elements is not the only possible sequence, because there are elements which may occur in many different positions in a group of other elements.

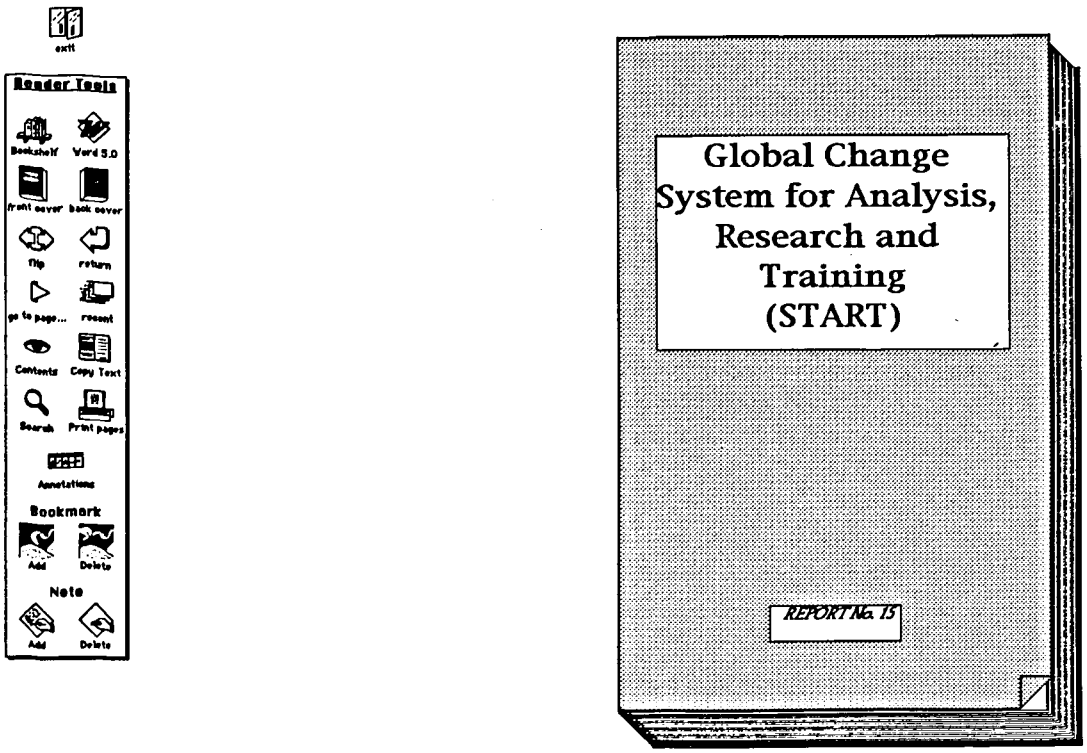


Figure 5.6: A closed book

The title page (see figure 5.7) and the verso page are the initial pages; each of them contains a number of components as described in section 5.1.1 (empty template). A number of other elements follows: abstract, foreword, preface, acknowledgements, etc. They always start from the right hand page, they have a title followed by text which may contain links, embedded figures and tables, and footnotes. The front matter also contains the Table of Contents (see figure 5.8), the List of Figures and the List of Tables. The Table of Contents is presented completely expanded, i.e. all the levels of the hierarchy are always visible. Each entry in the Table of Contents, List of Tables, List of Figures is active; clicking on one entry causes the book to be opened at the referred section, figure or table (hierarchical links). Each page in the front matter, except the blank pages, has a roman number.

The main body follows the front matter. Each chapter starts from the right hand page. A typical page in the main body may contain text, figures and tables which appear embedded in the text (see figure 5.9). The caption is shown below the table or the figure and has a smaller text size than the main text. Footnotes appear in overlapped windows only if the corresponding links have been activated. Text contains links which are indicated by a special HyperCard font style, group, as described in section 5.1.3.

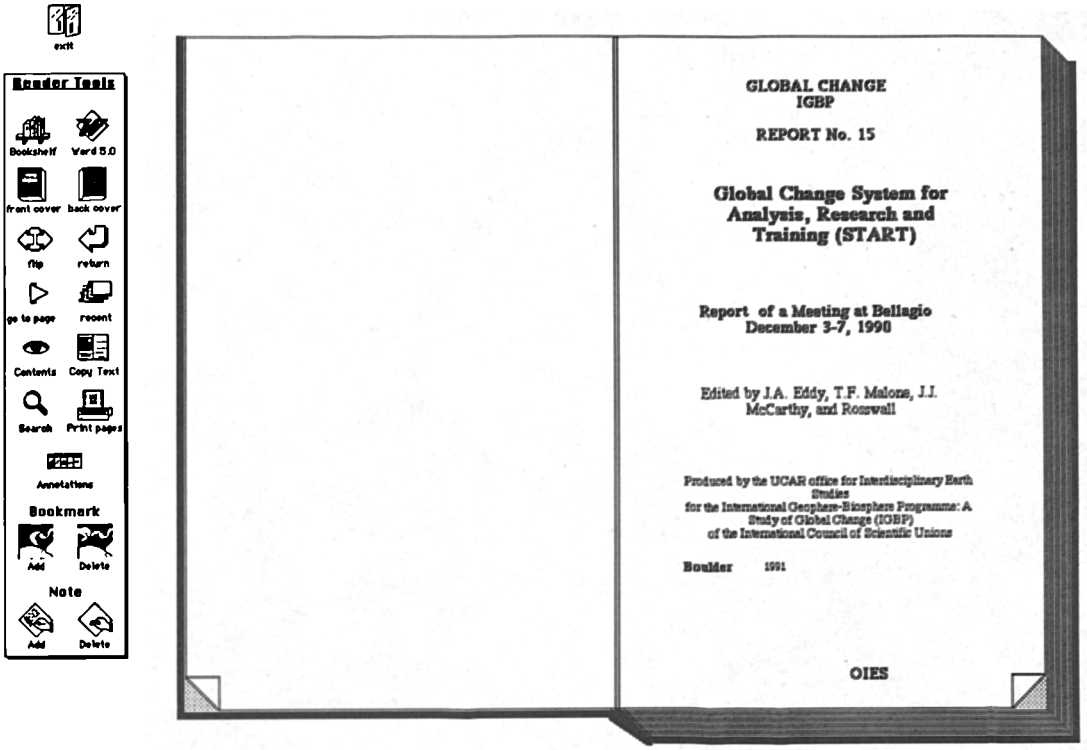


Figure 5.7: The title page

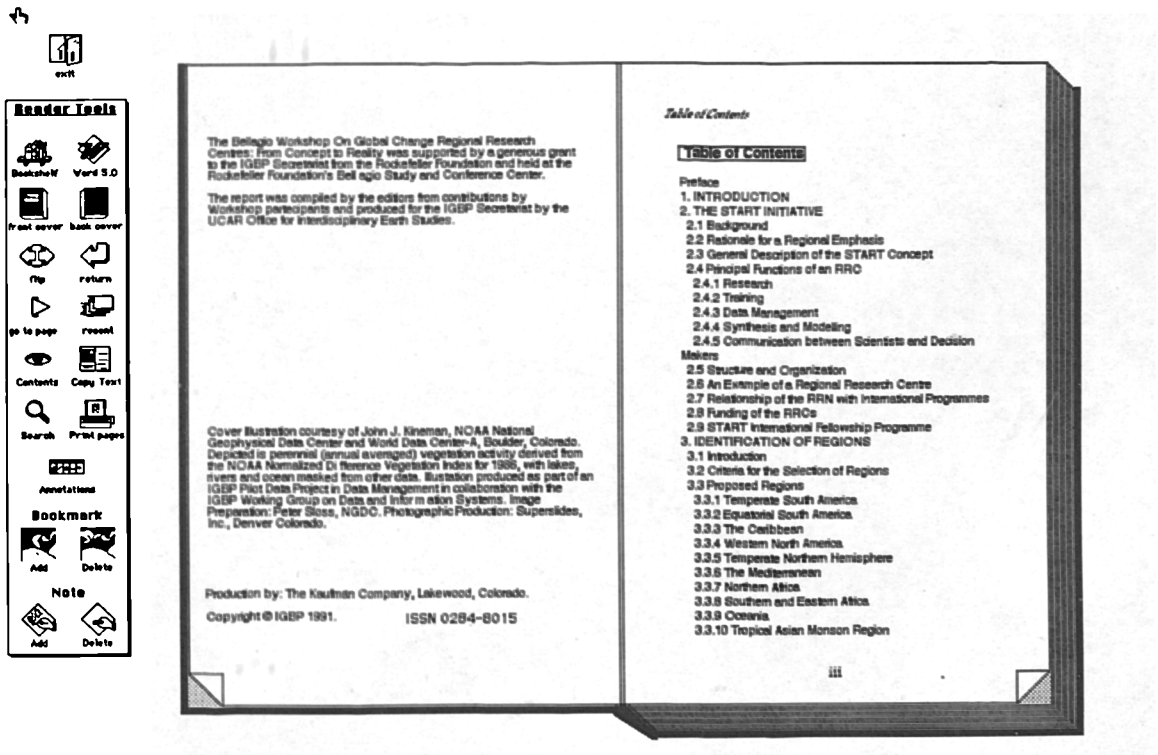


Figure 5.8: The Table of Contents page

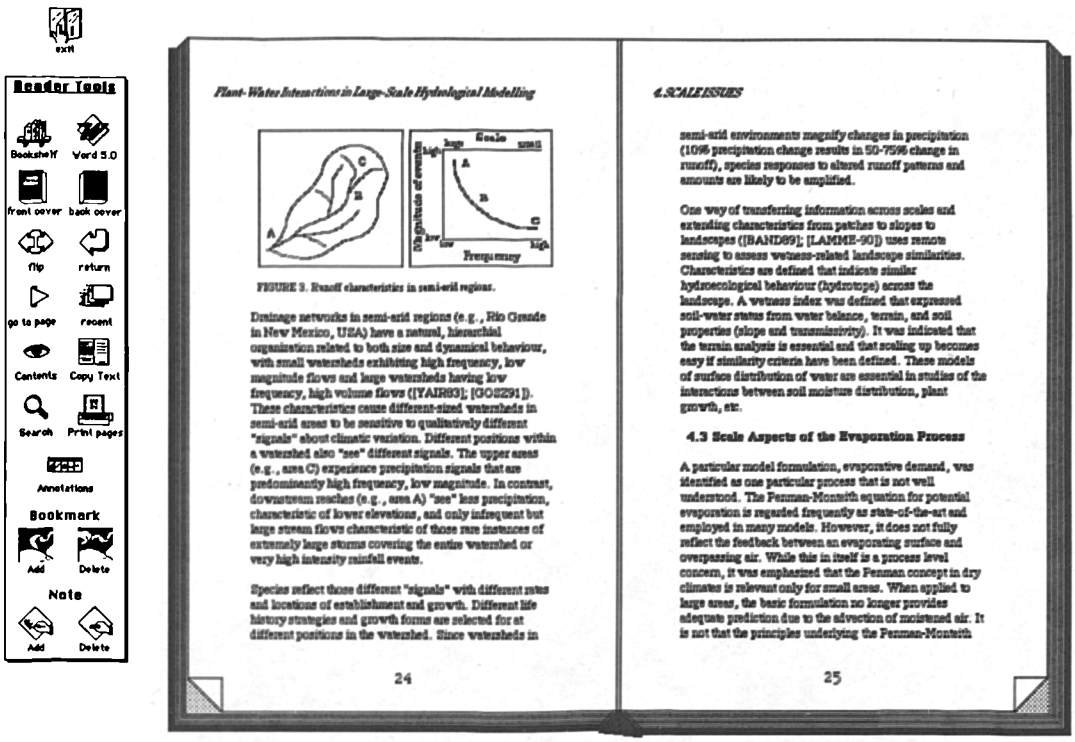


Figure 5.9: A typical page

The right header contains the title of the chapter, the left header contains the title of the book. The first page of each chapter does not display the header.

Each of the back matter elements (glossary, references, etc.) starts from the right hand page; important words (for instance acronym terms) are emphasised by using a different font or style. The index may contain two level terms (see figure 5.10); active logical identifiers replace the classical page numbers and indicate where the index terms occur in the book. The group style is used for these active identifiers.

The appendices are the last elements of the back matter and have the same appearance as the main body.

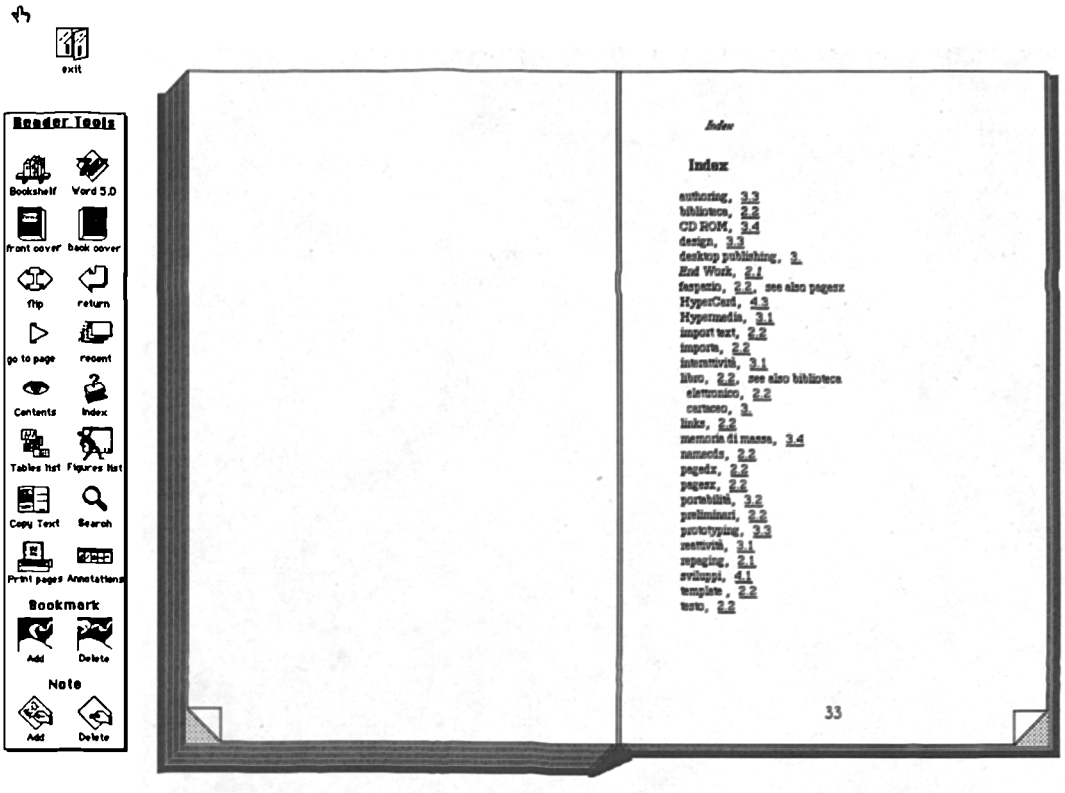


Figure 5.10: Index

5.2.2 Reader Services

Reader services are presented following the classification proposed in section 4.2.2 of the previous chapter.

- Navigation tools

They allow the reader to move inside and outside the book. Some of them can be activated directly from the book: they include the *next* and *previous* page represented by dog-ear buttons on the left-bottom and right-bottom corner of the book, respectively; the active *thickness*; the *link sources* in the text; and the active Table of Contents, List of Figures and List of Tables. The other navigation tools are represented by active buttons in the reader tool area.


The *flip* button: allows the user to flip right, if it is clicked on the right-hand side, or left, if it is clicked on the left-hand side. The mouse button needs to be held down in order to keep the flipping process active.

Other buttons provide direct access to important locations in the book: Table of Contents, Index, List of Figures and List of Tables, which represent a sort of natural browsing tool in the book. Their icons are displayed below:

 Contents

 Index

 List of Figures

 List of Tables

The  *Front Cover* and the  *Back Cover* buttons allows the user to close the book on the front cover or the back cover respectively. The book can then be opened by clicking the thickness, the next or back buttons, or the other navigation buttons.

 The *go to page* button opens a dialogue box where the reader has to enter a page number; if it is valid, i.e. in the range of page numbers available for that book, the book is then opened at the specified page, otherwise an error message appears.

As the *hyper-book* is a part of a collection of other books, a tool is provided in order to access such a collection. Because of the limited number of books considered in this practical development,

 The *book shelf* button simply shows a book shelf with the books currently available in the collection.

- Orientation tools

As described in section 4.2.2, the Table of Contents in the front matter (accessible using the Contents button), running heads, page numbers and the three-dimensional

representation of the book (thickness) together contribute to provide important orientation clues.

- **History tools**

There are two different ways of providing history mechanisms: the *recent* and the *return* buttons.


The *recent* button shows a series of miniatures of the most recently seen pages, up to 42; a click on any miniature allows the user to go to the page it represents.


The *return* button moves back to the previously visited page, which can be in the same or in another document. The return and recent functions are both available as built-in HyperCard commands.

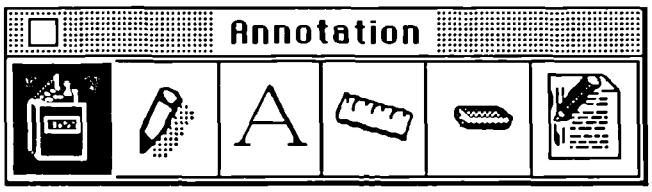
- **Personalisation tools**

Personalisation facilities include notes, which are written in overlapped windows, annotations, which are directly visible on the page and overlap the original page content, and bookmarks. Notes, annotations and bookmarks may be added and deleted using the following buttons.


A click on the *Add Notes* button makes the button flash showing that it is ready to create a note window; another click is necessary to indicate the location where the user wants the top left corner of the note to appear. The user is then asked to enter the name of the note which will appear on the title bar of the window. The note can be opened and closed by clicking on the small nail placed on the left side of the bar (see figure 5.11). The position of the note can be changed by dragging the bar.


A click on the *Delete Notes* button makes the button flash showing that it is ready to delete a note; another click is necessary on the small nail to indicate which note is to be deleted.

The *annotation* button (see figure 5.11) displays a palette which contains a number of other buttons for annotating the book (a free hand pen, a pen to type text, a line drawer, an eraser, and a big pen for marginal notes) and an additional button for selecting the browsing mode.



The *book* button allows the user to leave the annotation mode and to go back to the browsing mode, which is the default mode used during the book consultation.

The *pencil* button allows the user to add free-hand annotations. A click on the *pencil* button changes the cursor to a waiting state for a second click on the place where the user wants to start drawing.

The *graphic text* button allows the user to add text annotation, i.e. bit-mapped images of characters which can be deleted like other graphic annotations. A click on the *graphic text* button changes the cursor to a waiting state for a second click on the place where the user wants the insertion point to be set for typing.

The *line* button allows the user to draw lines. A click on the *line* button changes the cursor to a waiting state for a second click on the place where the user wants to start drawing a line.

The *eraser* button allows the user to erase existing annotations. The text of the book is never affected by these annotation tools. A click on the *eraser* button changes the cursor to a waiting state for a second click on the place where the user wants to start erasing by holding the mouse down.

The *big pen* button allows the user to mark interesting passages of text by drawing large lines, for instance along the margins. A click on the *big pen*

button changes the cursor to a waiting state for a second click on the place where the user wants to start drawing the large lines.

All these annotations can be inserted only in the book area. A highlighter, i.e. a coloured and *transparent* pen which allows portions of text to be marked, is not available in the current version. For practical reasons it could not be implemented though the *big pen* button partially replaces it.

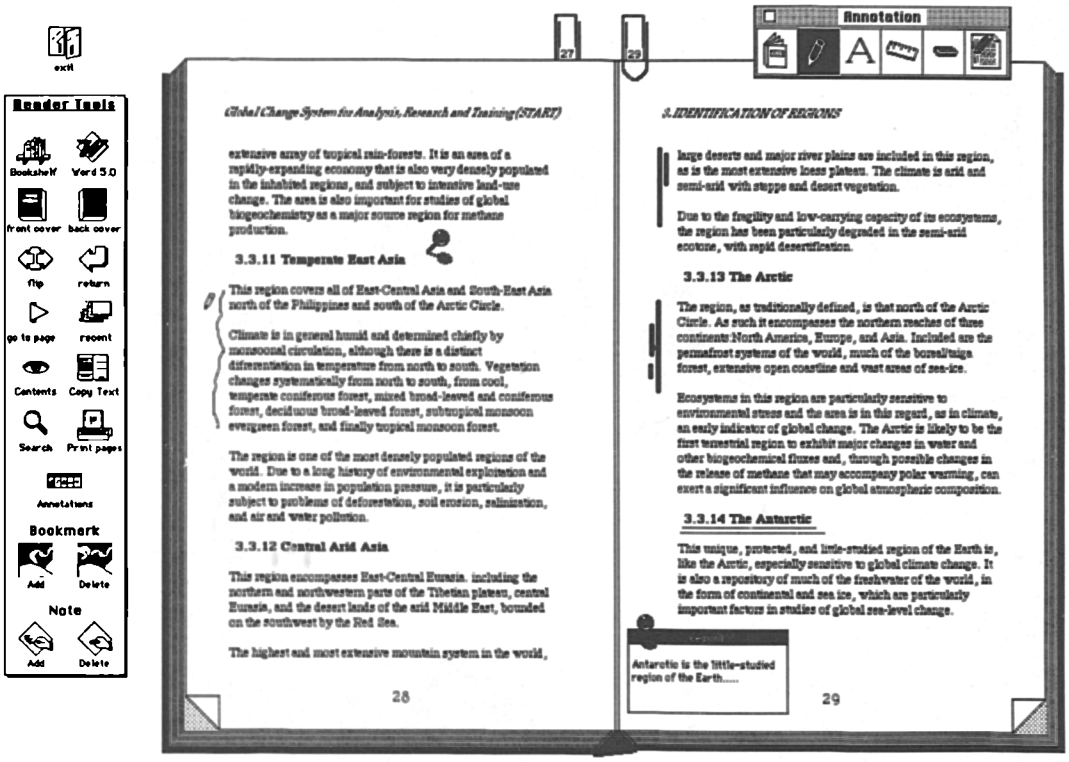


Figure 5.11: Personalisation tools

Bookmarks represent another tool which is offered to the user for personalising the book; they represent at the same time a backtracking or history mechanism. A bookmark is displayed as a small paper clip inserted on the top side of the book, with a number which indicates the page where it has been created. The choice of the top side of the book as a bookmark position and of the page number as a label for the bookmark was made for purely practical reasons (i.e. there were implementation difficulties associated with the limited space inside the bookmark).

 The *Add Bookmark* button automatically inserts a new bookmark on the right hand page. Bookmarks are placed in predefined positions starting from the centre of the book towards the right margin, leaving some space between the different bookmarks so that they cannot overlap. The width of the page represents a constraint for the number of bookmarks. When a new bookmark needs to be inserted, the first unoccupied position from the centre is chosen. Obviously when pages are turned, some bookmarks may need to be visualised on the left page: if the bookmark page number is less then the actual page number, then the bookmark is shown on the left page. Furthermore, when the bookmark number is different from the current page number, the upper part of the bookmark is visible, the other part is hidden by the page(s), as shown in figure 5.11.

 A click on the *Delete Bookmark* button erases the bookmark which is on the current page, if any bookmark exists.

- **Searching tools**

 The *search* button provides the capability of searching for a string or a specific word in the book. The user is asked to insert the search term; each occurrence of the search term in the text is shown one at a time. The user is presented with the first page which contains at an occurrence of the search term and has to press the "enter" key to view the next occurrence. The built-in HyperCard *find* function has been exploited.

There is a direct connection between the operators described in section 4.2.3.2. and the buttons and active areas described above. The *thickness*, *go to page*, and direct access buttons (i.e. *Contents*, *Index*, *List of Figures*, *List of Tables*, *Front Cover* and *Back Cover*) are based on the *direct-access* operator. The direct access buttons are special cases where the position index is fixed. The *next*, *previous*, and *flip* buttons are based on the *linear-access* operator. The link activation is based on the *hj-link* and *w-link* operators. The *recent* and *return* buttons are based on the *history* operator. The *return* button is a special case where the position index is fixed. All buttons which

represent personalisation tools are based on the *personalisation* operator. Finally, the *search* button is based on the *search* operator.

- External sources tools

When the user needs to read considerable parts of the book, the screen is certainly less appropriate than paper. For this reason a print tool is provided.


The *Print* button activates a dialogue box where the user is asked to choose between two options: print the whole book or print some pages; in the second case the user has to specify the range of pages s/he is interested in. The pages appear on paper as they are on the screen without showing the thickness and the reader tools. All the typographical information is maintained in the paper copy.


The *Copy text* button copies the currently selected text, if any, and puts it into the clipboard. The maximum selectable area is a page in the current version but it will be extended in the future to include more pages.

Readers are provided with the possibility of opening a word processor for writing their own notes and pasting portions of text which have been previously copied.


The *Word 5.0* button opens Microsoft Word 5.0; when the application is quitted, HyperCard gets back control.


The *Exit* button allows the user to quit the application.

5.2.3 How the Hyper-book Matches the Book Metaphor at the Physical Level

In order to complete the comparison between the *hyper-book* and the paper book, presented in the design chapter, this section considers the physical level. The physical level specifies the physical representation of the *hyper-book* on the screen and the actions performed by the users. *Hyper-books* are visually presented on the screen as paper books as widely described in section 5.2.1 and shown in the screen dumps.

The actions performed by the users of *hyper-books* are mainly mouse clicks on buttons and typing characters where required. While there is a good match between the *hyper-book* and the paper book for what concerns the representation of the interface, there is a complete mismatch with respect to the actions performed by the user. All the actions accomplished with paper books are related to their physical nature (e.g. physically turning a page) and had to be replaced in some way in *hyper-books* (e.g. mouse click in a book corner).

In the following tables, whenever there is a mismatch between the system domain (*hyper-book*) and the metaphor domain (paper book) and the system domain does not have a counterpart in the metaphor domain, then the cell is left empty. Otherwise, if there a mismatch but a corresponding action exists, then such action is described.

Physical Representation of the interface

HYPER-BOOK	PAPER BOOK
three-dimensional book representation (book space) + reader tools area	three-dimensional book
fixed book position on the screen (thickness visible always on the same sides)	changeable book position and inclination

Table 5.1: Physical representation

Users' Actions

HYPER-BOOK	PAPER BOOK
go to a specific page by clicking on the *go to page* button and entering a page number in a dialogue window	go to a specific page by browsing through the book (large "jumps" for approaching the page, flipping for getting the specific page)
open the book at the first page of the Table of Contents, Index, List of Figures, List of Tables by clicking respectively on the *Contents, Index, List of Figures, List of Tables* buttons in the reader tool area	find the Table of Contents, Index, List of Figures, List of Tables by browsing through the first pages of the book
open the book randomly by clicking on the book thickness	open the book by moving a block of pages to one side
close the book by clicking on the *front cover* or *back cover* button	close the book by moving the two blocks of pages together

Table 5.2: Navigation: Direct Internal Access

HYPER-BOOK	PAPER BOOK
access to the library by clicking on the *bookshelf* button	browsing books on shelves in the library
direct access to another document by clicking on an external link source	find a document referred to in the current book by searching through the library material or by using an inter-library loan

Table 5.3: Navigation: Direct External Access

HYPER-BOOK	PAPER BOOK
flip pages by holding down the *flip* button	flip pages by holding a block of pages and rifling through each page in the block
turn pages by clicking on the right or left dog-ear button in the bottom corners of the book	turn pages physically

Table 5.4: Navigation: Linear Access

HYPER-BOOK	PAPER BOOK
activate transverse links by clicking on a link source	find a referred object by browsing through the book
activate hierarchical links by clicking on a link source in the Table of Contents, List of Figures, List of Tables, and Index.	find a Table of Contents entry, a figure, a table, or an index entry by looking first for the page number in the Table of Contents, List of Figures, List of Tables, or Index and then looking for the specific page
zoom pictures/tables by clicking on them	

Table 5.5: Navigation: links

HYPER-BOOK	PAPER BOOK
open the book at the previously visited page by clicking on the *return* button	
show an active miniature of the last visited pages by clicking on the *recent* button	
open the book at a location where a bookmark has been inserted by clicking on the *bookmark* button	open the book at a location where a bookmark has been inserted by moving the block of pages before the bookmark on one side

Table 5.6: History

HYPER-BOOK	PAPER BOOK
start a full-text search by clicking on the *search* button in the reader tool area	search for a term in the Index

Table 5.7: Searching

HYPER-BOOK	PAPER BOOK
copy selected portions of text by clicking on the *Copy Text* button	copy selected portions of text by manually transcribing or photocopying them
open Microsoft Word 5.0 by clicking on Word *5.0* button	
print pages by clicking on the *print pages* button and entering the start and last page numbers in a dialogue window	
quit the system by clicking on the *exit* button	leave the book

Table 5.8: External Sources

HYPER-BOOK	PAPER BOOK
insert a bookmark by clicking on the *Add Bookmark* button	insert a bookmark by physically inserting an object or turning back the corner of a page
delete a bookmark by clicking on the *Delete* Bookmark button on the reader tool area	physically take out the object inserted as a bookmark from the book
add an overlapped empty window by clicking on the *Add Notes* button first in the reader tool area and then on a specific point on the book space in order to define the window position; click into the window before starting typing	add a note by sticking a paper note on the page of a book
delete an existing window by clicking on the *Delete Notes* button first in the reader tool area and then on the window which is intended to be deleted on the book space	take out a note by removing the paper note attached to a page
show the annotations palette by clicking on the *annotations* button in the reader tool area; activate a button on the annotation palette and drag the mouse on the book space in order to add or remove annotations	write and erase annotations by using a pen, a pencil, a highlighter or an eraser

Table 5.9: Personalisation

As Hammond and Allinson pointed out (Hammond and Allinson 1987), metaphors do not need to be appropriate at all levels of description; "a metaphor which is self-evidently inappropriate at some levels of description will be more successful than one which is ambiguous". The important aspect to consider is to avoid erroneous mapping between the two domains, because the user may be misled.

The conclusions which can be drawn from the comparison of the *hyper-book* and the paper book at the four levels of description presented in the previous (section 4.4) and current chapter, is that the *hyper-book* matches the book metaphor mainly at the task, semantic, and lexical level; the physical level is only partially covered: whereas the physical representation of the interface takes several aspects from the book metaphor, the actions performed by the user are completely different because of the different nature of the two objects (*hyper-books* and paper books). The *hyper-book* system may be helpful in communicating concepts to users, even if they cannot turn pages by stroking a touch-sensitive screen. However, if the action of turning pages were executed by stroking a touch-sensitive screen, other related actions (e.g. close the book) should be supported in the same way, rather than using a different strategy, such as clicking on a button.

Another aspect of metaphors is that they do not explain everything about the system (Hammond and Allinson 1987). In the *hyper-book* system, a number of facilities have been added, which are outside the book metaphor and which have improved the system.

5.3 Limitations of the prototype

HyperCard proved to be very effective mainly in the development of the prototype interface. However, it was sometimes inadequate for certain aspects of the system implementation. The main limitations are:

- limitation of the text formatting and managing capabilities:
  - text cannot be justified in the fields;
  - no provision of a pagination tool;
  - limited typographical capabilities;
  - difficulties in addressing specific words in the text, especially when the system dynamically changes (building process).

Some of these deficiencies have been solved by implementing the missing capability; for instance a pagination tool has been developed. Others, e.g. justification, would require an intervention on the HyperCard low level code, which is not accessible;

- system inefficiency: this is mainly due to the fact that HyperTalk is an interpreted language; therefore highly computational operations (e.g. importing text) often are quite slow. This is not considered to be a real problem as the purpose of the prototype is to prove functionalities rather than to be a commercially acceptable system. In addition, inefficiency affects the *hyper-book builder* more than *hyper-book* itself;

- limited image management capability: PICT images are not directly manageable in HyperCard except by using external commands. The result is that they are not well integrated in the HyperCard system

Most of these limitations will be solved as new HyperCard versions are released and other external commands are made available.

Other problems are intrinsically related to the implementation choices. The provision of the built-in HyperCard *find* function has avoided the need to develop a new indexing and searching tool; however this function has some limitations with respect to the design choices such as lack of feedback on the Table of Contents after a search and the fact that search terms are highlighted in the text one at a time.

Chapter 6:
Evaluation and Results

The previous chapters have described a flexible environment (the *hyper-book builder*) which allows electronic books (*hyper-books*) to be generated in a semi-automatic way. This environment provides the tools for acquiring and processing electronic texts in order to produce electronic books with an appropriate interface and reader services, based on the book metaphor. The focus of the evaluation is on evaluating the interface design and the reader services proposed in the *hyper-book* system. The evaluation of the *hyper-book* implies an indirect evaluation of the environment used for producing it. The evaluation of the editor tools developed for producing *hyper-books*, which are intended to be used by publishers, is not considered in this context. This is because the *hyper-book builder* was designed to demonstrate functionalities and prove concepts rather than to meet the commercial expectations of a publisher in terms of performance.

The evaluation has been constrained by the general context within which this research has been conducted (the Super-Library project) as the application has been developed within the framework of that project. A set of documents in the environmental field was selected for use as a test application. A group of potential users was also selected who have an interest in the field of these documents.

In this chapter, a brief review of the literature in the evaluation field is provided first, in order to summarise the main concepts and available techniques. Secondly, the objectives of the evaluation and the procedures followed during the evaluation are discussed. A significant part is dedicated to the description of the test application. Finally, the results and the main conclusions drawn from the evaluation are reported.

6.1 Evaluation: Literature Review

"Evaluation is concerned with gathering information about the usability or potential usability of a system in order either to improve features within an interface and its supporting material or to assess a completed interface" (Benyon at al. 1990). An evaluation has two main objectives:

- to determine the effectiveness of an interface in use;
- to provide a means for suggesting improvements.

Evaluation is a process strictly connected with the design and development cycle of a system and can occur at different stages: formative evaluation takes place during the development phase of a system, summative evaluation takes place in order to evaluate the final system.

A system which is under development should be modified on the basis of the results of the evaluation. The process must be repeated to iteratively improve the system; this is called iterative design (Gould et al. 1991).

This session presents a brief review of the main techniques used for evaluating a system, introduces the main usability parameters and mentions some examples of evaluation of working systems.

6.1.1 Evaluation Techniques

It is possible to identify different methods for evaluating a system. In particular Benyon et al. (1990) distinguish between five categories:

- analytic evaluation
- expert evaluation
- observational evaluation
- survey evaluation
- experimental evaluation

Analytic evaluation uses a formal or semi-formal interface description to predict users' performance in terms of the physical and cognitive operations that must be carried out. It can be used early in the design cycle because it requires few resources, it does not require costly prototype and user testing. However, the focus is quite narrow and it does not produce a diagnostic output for redesign; in addition, it is based on broad assumptions concerning the users' cognitive operations.

In *expert evaluation* experts in interface design or HCI, are asked to judge the system and identify the potential usability problems for less experienced users. This method is quite cheap and efficient compared to methods which involve user testing: a small number of experts can detect significant problems. However, experts have to be chosen with caution in order to avoid biases and to capture real user behaviour. This method can be used on early prototypes or system specifications. Expert evaluation has also been called heuristic evaluation (Nielsen and Molich 1990) in contrast with empirical evaluation, which involves experiments with test users. Observational and survey techniques are mainly used in this last form of testing (empirical evaluation).

The purpose of *observational evaluation* (Benyon et al. 1990, Gomoll 1990) is to collect data about the user's behaviour while using a system. Several techniques can be used; the most relevant are:

- direct observation: the evaluator makes notes of the user's actions;
- video recording: the user's activity is recorded and then replayed with or without his/her participation;
- software logging: this is an unobtrusive method; user interaction with the system is automatically recorded;
- verbal protocols (thinking aloud): the user is invited to think aloud to express observations and thoughts while using the system. From this method a wide range of information can be obtained concerning for instance the user's planning for the particular task, and recall of commands. However it can be difficult for a user to express his/her thoughts when s/he is trying to solve a difficult problem.

These techniques, with the exception of software logging, are obtrusive methods; therefore they may affect user activity and performance.

The data analysis, which follows the data collection, can be time-consuming and resource-consuming, but can provide interesting qualitative data. Usually, this phase is important to determine how the user faces tasks, where the main difficulties lie, and to obtain performance measures such as task timing, use of commands, frequency of the user's errors, etc. This method is generally used with prototypes which have reached a good stage of development.

Survey evaluation involves the use of interviews or questionnaires with the purpose of eliciting users' subjective opinions and understanding of the interface. Interviews can be flexible (a set of general topics is defined but not a precise sequence), or structured (a sequence of questions is predefined). Questionnaires can provide two

main classes of questions: closed questions, when users have to choose from a set of proposed alternatives, and open questions, when users are free to provide any answer. Open questions may be more difficult to analyse but provide a richer source when compared to closed questions.

Interviews are normally used with a restricted group of users and can be very time-consuming; questionnaires can be used with very large groups of users. A detailed analysis of this technique (questionnaires) is reported in (Dautriat 1990).

A valuable example of questionnaire for evaluating user interface has been published in (Shneiderman 1992). The questionnaire is very general, but could be adapted to the a specific system by adding domain specific questions and terms.

In an *experimental evaluation* the evaluator can manipulate a number of factors associated with the interface design and study their effects on various aspects of users' performance. This requires a good knowledge of experimental methods and is very resource and time-consuming. Usually this method is applied to a fully developed prototype.

The choice of a particular method depends on the stage of development of the interface, the extent and type of users' involvement, the kind of data expected, external limitations such as time constraints, cost and availability of equipment, etc.

With methods which involve real users, it is important to choose subjects which represent the user community. Important factors are background, experience with the task, motivation, etc. Subjects should be informed that they are not under test, but it is the system that is being tested; they should be told what they are required to do and how long it will take (Shneiderman 1992).

6.1.2 Usability

Usability is the question of how well users can exploit the system's functionality (Nielsen 1990). Usability testing aims to assess how usable a system is going to be.

The concept of usability has been widely discussed and some usability parameters have been identified. It is possible to find various definitions in the present literature: Nielsen, for instance, considers usability in terms of:

- ease of learning (the user can quickly get some work done with the system);
- efficiency of use (a high level of productivity is possible);

- memorability (when the user returns to the system, he is able to use it without having to learn everything again);
- number of errors (users do not make many errors);
- how pleasant it is to use (users are subjectively satisfied by using the system).

Others (Benyon et al. 1990) identify as key aspects of usability the following:

- learnability (i.e. time and effort required to reach a specified level of user performance);
- throughput (speed of task execution, errors made, etc.);
- flexibility (the extent to which users can adapt a system to new ways of interaction as they become more experienced);
- attitude (the positive attitude engendered in users by the system).

The two definitions have many commonalties: for instance, ease of learning and user satisfaction are present in both definitions. The choice of the parameters have to be adapted to the specific situation and objectives of the evaluation.

These general parameters have to be measured in some way. For learning ability, typical measure are completion time for a specific task, the number of errors per task, the number of features used, etc.

For more information about usability see (Whiteside et al. 1988, Sullivan 1989).

6.1.3 Examples

There are essentially two ways of evaluating a system: *benchmark studies*, where various approaches are compared, *non-benchmark studies*, which avoid measurable comparisons of two or more conditions, because either a single system has been tested or because qualitative evaluations are considered more appropriate. A survey of benchmark research is reported in (Nielsen 1990). The experiments mentioned refer to the evaluation of several approaches to accessing text online.

A common approach is a comparison of the electronic version with the corresponding paper version; this approach is usually adopted when the purpose is to demonstrate that one representation is better than the other and by how much. Typical examples are the evaluation of SuperBook (Egan et al. 1989a, Landauer et al. 1993), and Sherlock Holmes stories (Mynatt et al. 1992).

Sometimes the comparison takes into account more than two representations, for instance different electronic representations (e.g. hypertext, scrolling text files, databases, traditional menu selection systems) or/and the paper version. Typical examples are the Core project (Lesk 1991), the Compton Encyclopaedia (Large et al. 1992), Grolier's Academic Encyclopaedia (Marchionini 1989). Another valuable

example is reported in (Rada and Murphy 1992), where different representations of the same text in paper and electronic form are compared and evaluated using three classes of users: experts, novices, and trainers.

6.2 Hyper-book Evaluation

The analysis of the literature in the area of evaluation techniques and discussions with experts in the field have been effective for the definition of a suitable method for evaluating the *hyper-book* system. This section clarifies the main objectives of the evaluation, introduces the test application, and describes how the *hyper-book* evaluation has been conducted.

6.2.1 Objectives

As mentioned in the introduction of this chapter, the focus of the evaluation is on evaluating the interface design and the reader services proposed in the *hyper-book* system.

Within this context, a set of objectives has been identified:

- verifying that the *hyper-book* system is easy to use and understand;
- verifying that the *hyper-book* system can be understood and used to a similar degree by everyone, regardless of his/her computer expertise;
- assessing users' opinion about the system utility and usability; in other words, it is not enough to demonstrate that the system is easy to use; other relevant questions are how useful is the system considered to be by its potential users, do users appreciate the interface design, what functionalities does the system provide, etc.
- collecting information and suggestions for improving the current prototype.

These objectives contribute to demonstrating the appropriateness of using the book metaphor for presenting electronic books.

6.2.2 Methodology

Having established the objectives, an important aspect of the *hyper-book* evaluation was the choice of the evaluation technique. On the basis of the analysis presented in section 6.1.1, and considering the current stage of development and the constraints

imposed by the research context (e.g. cost, availability of equipment, time, etc.), the most appropriate way for evaluating the system emerged as being an empirical evaluation, i.e. a evaluation which involves a group of potential users. A combination of observational and survey techniques was employed. In particular, software logging and direct observation were used for collecting data about users' behaviour while using the system; and questionnaires were employed in order to elicit users' opinions about the *hyper-book* system.

In order to verify that the *hyper-book* system can be understood and used to a similar degree by everyone, regardless of their computer expertise, two groups of users were considered, each of which had different experience with computers. The two groups were classified as computer experts and non-experts. Their performance was compared in terms of executing a number of tasks, which are discussed in section 6.2.4.

Taking into account the current stage of development and the objectives mentioned above, the evaluation of the *hyper-book* itself, rather than a comparison with other forms of organising and presenting electronic text, was considered appropriate. A comparison with other systems could be an interesting subject for future investigation but it has not been conducted in this research for a number of reasons:

- the costs of buying a product which can be compared to the *hyper-book* system (same purpose, similar functionalities, etc.);
- the need to present the same set of documents within each product; this could require the study of the new system, and the conversion of the original source text into another format if the new system accepts another kind of mark-up language.
- it may not make sense to compare a prototype with a commercially available system.

6.2.3 Test Application

This section first introduces the area of the test application (*Global Change*) giving a general description and mentioning the main concerns and problems involved. The second part explains the rationale for the choice of the documents. It also describes the preprocessing phase, which was necessary to produce documents in the format required by the *hyper-book* builder.

6.2.3.1 Application Area: Global Change

The term *Global Change* embraces a number of complex processes effecting the Earth and the people on it (IIASA 1991). They cover global phenomena such as population growth and urbanisation (which resulted in high economic expectations), the spread of new technologies and the Third Industrial Revolution, and patterns of energy use and their effects on climate. These processes are occurring on a global scale and no nation alone can control them (IGBP 1992). The Earth system is under severe stress: the atmospheric concentrations of several trace gases is increasingly changing the radioactive properties of the atmosphere. Serious environmental problems arise such as the green house effect, the depletion of the stratospheric ozone layer, and acid deposition. Therefore, there is an urgent need to expand scientific understanding of the causes and of the potential consequences of global trends, and to develop responses to them.

Interest in the *Global Change* topic is growing rapidly. The world scientific community and policy makers have been involved and encouraged to collaborate in formulating an international policy on the emerging issues of *Global Change*. Conferences, seminars, workshops have been organised all over the world; several programmes have been undertaken: the World Climate Research Programme (WCRP), the International Geosphere-Biosphere Programme (IGBP), the Global Environmental Research Programme. Many publications (reports, newsletters, proceedings) are now available in the area of *Global Change*.

The *Global Change IGBP* Series has been selected in order to be used as a test application for *hyper-book* evaluation. The International Geosphere-Biosphere Programme (IGBP) was established in 1986 by the International Council of Scientific Unions. The objective of the IGBP is to "describe and understand the interactive, physical, chemical, and biological processes that regulate the total Earth System, the unique environment that it provides for life, the changes that are occurring in this system, and the manner by which they are influenced by the human action" (IGBP 1991).

Two reports have been chosen as being representative of the series: Report 15 (IGBP 1991), *Global Change* System for Analysis, Research and Training (START) and Report 17 (IGBP 1991a), Plant-Water Interactions in Large-Scale Hydrological Modelling.

Report 15 contains the results of a meeting held in Bellagio in December 1990. The purpose of the meeting was the development of a plan for promoting an international research effort, in which the scientific communities in developing countries are fully involved. The plan has been called the START initiative. START is a system of interconnected regional research networks with the purpose to enhance the effectiveness of national research on the regional origins and impacts of global environmental change, and capacity-building through training and fellowship schemes (IGBP 1991). To implement START, a number of regions have been defined. "A region should be sufficiently large to embrace areas which are generally homogeneous and representative in terms of vegetation and climate; at the same time they should ideally include biogeographical diversity adequate to allow gradient studies" (IGBP 1991). The START project is still underway and its developments are reported in the *Global Change* Newsletter.

Report 15 is not a technical report and can be easily understood by a reader who does not have a specific knowledge of the general topic. Furthermore, it is a good example of the class of documents described by the *hyper-book DTD*: it has a multi-layered hierarchical structure, it contains acronyms, links, etc. This makes the report particularly appropriate for the evaluation phase.

Report 17 summarises the scientific issues addressed at the workshop in Vadstena (Sweden) in June 1990. The main topics discussed include: " (i) The spatial pattern at landscape level of the dynamics of water flows and waterborne fluxes of dissolved and suspended matter; (ii) plant/vegetation characteristics and regolith properties affecting return flow to the atmosphere, in particular, water-use by vegetation; (iii) observational and scaling-up methodological issues to support large-scale modelling and; (iv) plans for focused research in three major hydroclimatic regions: humid tropical, semi-arid, and temperate zones" (IGBP 1991a). One of the major challenges is solving scaling incongruities which exist between atmospheric, hydrologic, and terrestrial components of the Earth System. Numerical models, which describe, simulate and predict behaviour of ecohydrologic processes, are required in order to integrate the different scales.

Report 17 has been considered a suitable document for the test application because it is quite representative of the general class of documents described by the *hyper-book DTD*: it has a multi-leyered hierarchical structure, it contains several figures and tables, and it provides references and some kinds of links (g, xe, b) which are not contained in Report 15.

6.2.3.2 IGBP Reports in the Hyper-book Model

The choice of this series has been constrained by the more general project within which this research has been conducted. The same documents also had to be used for another research project (Landoni et al. 1993) which involves the conversion of paper documents into electronic documents which maintain the same visual aspects of the originals.

The IGBP collection presents a number of advantages which justify its usage as a test application for this research. The books are quite small in size and page numbers; therefore not much effort and time has been required in order to organise them according to the *hyper-book* model. Nevertheless, their features are sufficient in order to demonstrate the main objectives of this research. They contain text, graphics and tables and their logical structure can be easily assimilated in the book structure described in the *hyper-book DTD*.

A number of practical issues arose in the implementation of the test application. The IGPB collection is published in paper form, whereas the *hyper-book builder* requires documents marked in SGML as input. Therefore, first of all, it was necessary to have the electronic version of the documents and then to mark them. Pictures and tables were acquired directly from the paper version using a scanner. The electronic version of the text was provided by the IGBP Segretariat in Stockholm[1]. The text, originally available in WordPerfect form, was then reduced to ASCII form: information about the layout was not required in the conversion process and an automatic converter from WordPerfect is not supported by the current version of the *hyper-book builder*. The documents were then marked up: the mark up process has been carried out manually, inserting tags directly in the text. A manual process had to be employed because no SGML editor was available. However, the amount of text to mark up was not so large to make the use of an editor indispensable.

A parser was then used to verify that the tags were syntactically correct; then the automatic translation was activated (see section 5.1.2). Both the parsing and translation processes were carried out using the Exoterica's XGML translator. Few changes of the electronic book format were made in order to produce the final version.

[1] No copyright problem arose as the publisher gave permission to use the documents for research purpose.

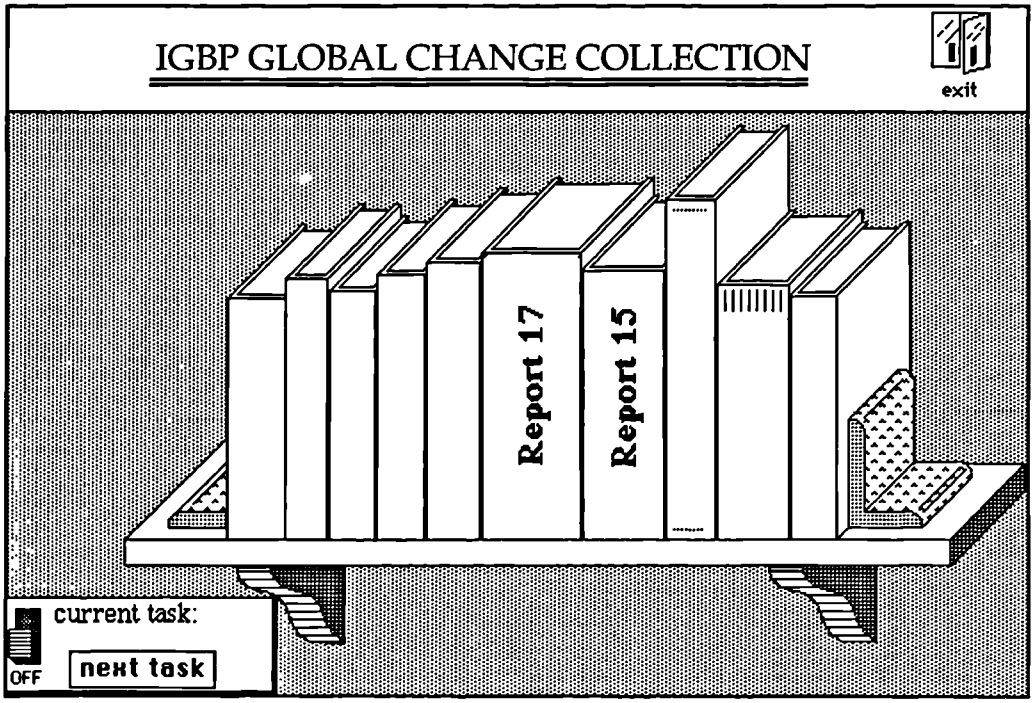


Figure 6.1: The bookshelf

The total time required for the preprocessing and semi-automatic conversion was about two weeks. Additional effort was dedicated to the definition of a basic library environment, a bookshelf with the two active reports (see figure 6.1) and to the incorporation of the logging routines which were necessary for the evaluation phase. The rectangle located at the bottom-left corner of figure 6.1 is used excluvely within the evaluation in order to control the logging process: the "next task" button allow the activation of a new task to be recorded in the log file, and the switch button allows the logging process to be interrupted at any time and, successively, restarted.

Examples of displays from the IGBP test application can be found in the previous chapter.

6.2.4 Procedure

The experiment consisted of several steps. Firstly, users were asked to fill in a brief questionnaire about their prior experience. Secondly, a training session was carried out in order to get the user acquainted with the *hyper-book* system. Thirdly, an evaluation session followed the training session, for the assessment of the system usability. Finally, users were presented with another questionnaire in order to assess their opinion about the system usability and utility.

The test application as well as the questionnaires and the task descriptions were all in English.

6.2.4.1 Initial Questionnaire

Twenty subjects were asked to fill out a brief questionnaire about their background, their computer experience, their familiarity with scientific English and their knowledge about the subject of the books (*Global Change*). The questionnaire was useful in order to select the subjects for the evaluation. A minimum requirement for being admitted was that the subjects were able to understand the questions of the tasks assigned. Therefore, the subjects were required to have a reasonable knowledge of scientific English and some knowledge of the subject area (*Global Change*), but not of the specific documents used as a test application in order to prevent the results from being invalidated by prior knowledge of the topic. Another requirement was that subjects had at least a limited experience with a computer. In fact, it is assumed that the majority of users of scientific publications (which represent the intended audience for this kind of electronic book) have some kind of familiarity with computers; therefore, the evaluation did not include people who had no experience. Computer familiarity was an important factor in order to distinguish computer experts from computer non-experts.

The full contents of the initial questionnaire are reported in Appendix B.

6.2.4.2 Training Session

Users were given a brief introduction to the system concepts and use. They were then asked to use the system without a specific task to accomplish. The users were invited to try out any function of the *hyper-book* in order to get familiar with the system and to ask for clarification, if necessary. The training was carried out using an electronic book which was not part of the collection used in the evaluation session. Thus, users would not learn anything about specific contents of the test application during this session. Each training session lasted about 15 minutes.

At the end of the training session, users were given a brief introduction to the tasks they were going to accomplish, what they were about, in which form answers were required, and other details about the evaluation session.

6.2.4.3 Evaluation Session

In this session users were given a number of tasks to accomplish, in order to assess the system usability. Few differences were expected between the two groups because the system should be easy to understand and use for everyone independently of his/her computer expertise. Nevertheless, some differences could arise for some aspects of the system which are not present in the paper book (e.g. searching facilities, links, etc.). This is expected because computer experts are already familiar with them and could use them more frequently than non-experts.

A set of tasks was chosen which was representative of typical tasks a user performs with books in a library. The basic ideas have been taken from the CORE Project, which presents some similarities with the *hyper-book* project because both are based on the concept of an electronic library (see section 2.5.2.1). General tasks such as browsing and searching have been adopted also for the *hyper-book* evaluation. Other tasks (e.g. analogous transformation) have not been considered as they were strictly dependent on the CORE domain. For additional details about the tasks used in the CORE evaluation, see (Lesk 1991, Egan et al. 1991a). The tasks chosen for the *hyper-book* evaluation can be grouped into three categories according to the classification proposed in section 4.4. These categories include:

- *browsing task*: given a general topic, the user wants to find out if and where the topic is discussed in the collection. This can be the case of a user who is browsing along a shelf of books looking for material on a topic. This task requires ability to consult a book in order to have a general idea of its contents.

- *searching task:* the user has a specific question in mind, whose answer is contained in a particular book. This is usually the case of a user who has some knowledge of the book's content and wants to retrieve information which is contained in it.

- *analysis task:* the user has to analyse a topic discussed in the collection and select the most relevant parts. This could be a typical problem of a researcher who has to use some documents as a reference material for a paper. What is more interesting for the purpose of this evaluation is to investigate which tools are mostly used during the analysis rather than how the user performs. The user is free to choose any tool.

Topics and questions were formulated with experts in the field of *Global Change* in order to guarantee reasonable and adequate choices of tasks. Usually the questions do not require a deep knowledge of the area. The specific topics and questions are reported in Appendix B. During the formulation of topics and questions, a set of correct (predefined) answers were defined. For the browsing category, four tasks were considered; for the searching category, ten tasks were chosen; for the analysis task, just one task was indicated.

For each task, four aspects were considered:

- how well the task was accomplished (quality);
- the time required to accomplish the task;
- which tools were used and how often;
- where the main difficulties were encountered.

The task quality was evaluated across a range of four possible scores:

correct, if the answer was correct,
partially correct, if the answer was partially correct,
incorrect, if the answer was wrong,
out of time, if no answer was given in the time assigned.

A relatively coarse granularity, i.e. a low discrimination of the answer quality (correctness), was used because of the nature of the questions. Some questions required a yes or no answer, others required a precise answer (an acronym, a name). For those answers, the possible scores could be just correct and incorrect. For more complex questions, an intermediate score was necessary (see questions in Appendix B). Moreover, incorrect answers were distinguished from answers not given in the assigned time.

The time needed, as well as the tool used, to execute a task was derived from the log file.

The usability difficulties were identified by inviting users to express their difficulties while using the system.

The quality measures provide an indication of how correctly and consistently the information is displayed. The time measures provide an indication of how easily the

users found it to perform each task. The better the quality of answers , the better the system is. The shorter the time required to accomplish tasks, the easier it is to use the system.

The role of the interviewer in this phase was to observe the user and to take notes of his/her comments and difficulties, without taking over or giving instructions.

Log files were used during this session for recording the user's actions. An excerpt from a log file is given below. Each task was given an identifier (e.g. task 2.8), which was recorded in the log file, together with the time when the user started to execute it (16:27). A history of the user's actions - an action corresponds to the activation of a tool - was then recorded in terms of tool name (e.g. toc, nex, se, etc. in the example below) and the identifier of the page where the tool was activated (e.g. 14016, 2290, etc.). In some cases, e.g. transverse links and search, additional information, such as the link source (e.g. IGBP) and the searched term (e.g. SC-IGBP), respectively, was also recorded.

```
task 2.8,16:27

toc,14016
nex,2290
nex,4524
TL,4823,IGBP
pre,4823
nex,4524
nex,4823
nex,5004
se,5203,SC-IGBP
nex,5584
```

The user was given a maximum time for accomplishing each task: four minutes for browsing and searching tasks, ten minutes for the analysis task. This time was estimated to be sufficient to execute the given tasks, taking into account the results of the pilot study, mentioned below.

6.2.4.4 Final Questionnaire

Users were required to answer a number of questions about the usability and utility of the system. As discussed in section 6.1, there are different usability parameters. Obviously, they have to be adapted to the specific case. Within the context of this research, the most important parameters in order to measure the *usability* of the *hyper-book* are:

- *ease of learning*, which refers to how easy it is to understand the system quickly and without lengthy training. The objective of using the book metaphor is to guarantee a high degree of familiarity with most of the system aspects;

- *error frequency*, which refers to the errors the user makes while using the system because of system misunderstandings. Typical errors can be, for example, erroneously following a link or not being able to return to an already visited location. Few errors are expected in systems which are unambiguous and easy to understand;

- *how pleasant it is to use*, which indicates the degree of satisfaction users can get from using the system. A high value of this parameter denotes that users rarely feel frustrated or disappointed, that they are in control of the system, and can move freely.

Utility is the question of whether the functionalities provided by the system are really useful for achieving the desired goal. In other words, if two sets are considered: users' needs and system functionalities, the requirement is that the system functionalities are enough to cover the users' needs.

6.2.5 Execution of the Evaluation

Before executing the evaluation, a pilot study was carried out in order to refine the details of the procedure described below (e.g. questions, topics, maximum time). A pilot study is generally used for determining the suitability of an experimental design before starting the real evaluation. The pilot study consisted of the execution of the evaluation on a number of "pilot" users. These users had considerably contributed to the definition of an acceptable protocol for evaluating the *hyper-book* system.

Then, the real evaluation was executed. The subjects which participated in the evaluation were all researchers at the EEC Joint Research Centre in Ispra. The evaluation was conducted in a small office, one user at a time. The duration of the evaluation ranged approximately from one hour to one hour and a half.

In the following section the results of the evaluation are summarised and elaborated in order to assess the *hyper-book* model.

6.3. Results of the Evaluation

The results are presented according to the procedure adopted in the evaluation. A number of bar charts and tables help to synthesise and compare the performances of the two groups. The full results are reported in Appendix C. Observations and conclusions are contained in the next section.

6.3.1 Results from the Initial Questionnaire: Subject Selection

The initial questionnaire was useful in order to select the subjects for the evaluation. Subjects were selected according to the requirements described in the procedure of the evaluation (Initial Questionnaire, section 6.2.4.1). All subjects had some knowledge of the domain of the documents used for the test application (*Global Change*), but they were not familiar with the original paper version. In addition, they had reasonable knowledge of scientific English. However, whenever a problem of English comprehension occurred during the evaluation, limited help from the interviewer was allowed.

On the basis of the answers which were related to their experience with computers, subjects were divided into two groups of 6 people each, computer experts and computer non-experts. Group 1, computer experts, consisted of people with a background in computer science, who spent most of their working day using computers, and who were highly familiar with Apple Macintosh machines. Group 2 consisted of people with a background in biology or physics, who sometimes used computers and who were familiar with a small number of programs, mainly word processors.

For reasons of homogeneity the size of the groups was quite small. However, this number was considered sufficient at this stage of development, because the evaluation had an exploratory purpose and was intended to suggest improvements for the current prototype, which is part of an iterative design cycle.

6.3.2 Results from the Training Session

The subjects selected by means of the initial questionnaire, were then required to participate in the training session. The experiment has demonstrated that a short time (15 minutes) is sufficient for people to get familiar with the system. All subjects were

later able to use the system confidently, as indicated by the results of the evaluation session.

6.3.3 Results from the Evaluation Session

This section reports the results of the evaluation session. Three aspects will be considered: quality of the users' answers for tasks, time taken to accomplish a task, and frequency of tool usage. Few difficulties were identified during the observation of users performing their tasks. The problems which emerged in this phase were the same as those indicated later in the answers of the final questionnaire. See section 6.3.4.1 for more details.

6.3.3.1 Quality of Answers

This section presents the quality of the users' answers to the assigned tasks. The correctness of the answers was measured by comparing the given and the correct (predefined) answer. The assessment of the answer quality was trivial when the answer expected was yes or no. Otherwise a matching was accomplished between the given and the predefined answer by carefully interpreting user's responses.

Table 6.1 shows a comparison between the frequency of answers given by the two groups of users for the *browsing* task across the range of the four possible scores. As usual, Group 1 indicates computer experts, Group 2 indicates computer non-experts. Each frequency value respectively indicates the percentage of correct, partially correct, incorrect, and out of time answers scored by each group.

Browsing %	Group 1 (N=6)	Group 2 (N=6)
Correct	58	63
Partially Correct	17	8
Incorrect	17	29
Out of time	8	0

Table 6.1: Answer type frequency in the browsing task

Firstly, it is possible to observe that the majority of answers in both groups are of correct type and no substantial difference can be seen between the two groups. Secondly, the low frequency of out of time answers combined with the good answer

quality indicates that the system can be easily used for achieving good results by both groups.

In general, it is possible to note that there is a slightly better performance of computer non-experts, concerning both the correct and out of time answers; this could be attributed to the fact that computer experts preferred to use searching rather than navigation tools (as better described in section 6.3.3.3), whereas for the browsing task, navigation tools could have been more appropriate.

Analogously to the previous table, table 6.2 shows a comparison between the frequency of answers given by the two groups of users for the *searching* task across the range the four possible scores.

Searching %	Group 1 (N=6)	Group 2 (N=6)
Correct	77	62
Partially Correct	10	23
Incorrect	10	12
Out of time	3	3

Table 6.2: Answer type frequency in the searching task

For the searching task, the majority of answers in both groups are correct and no substantial difference can be seen between the two groups. In addition, the frequency of out of time answers is very low for both groups. These two findings confirm the already observed result that the system can be easily used for achieving good results by both groups.

In general, it is possible to note that there is a better performance from computer experts; this could be attributed to the fact that computer experts are more used to employ searching tools, which are mainly required by this task.

Additional conclusions can be derived from a further elaboration of the answer quality. Assuming that the difference between correct, partially correct, and incorrect answers is 1, and that incorrect and out of time answers have the same weight, a value, ranging from 0 to 2, is attributed to the answer quality, according to the following criteria:

2, for correct answers
1, for partially correct answers, even out of time
0, for incorrect answers, or no answer in time

Assuming that:

$UserQuality_{k,i}$ = the score assigned to the answer of question k given by user i

an average quality in each group (*GroupQuality*) was computed over the scores of the users in that group:

$$GroupQuality_k = \frac{\sum_{i=1}^{N} UserQuality_{k,i}}{N}$$ where N is number of subjects in the group

Both *UserQuality* and *GroupQuality* are relative to a single question. *GroupQualities* are the quality values shown in the following charts. Another parameter which is interesting to consider is the *GlobalGroupQuality*, which represents the average answer quality for a given group over the set of questions assigned for a certain task:

$$GlobalGroupQuality = \frac{\sum_{k=1}^{M} GroupQuality_k}{M}$$ where M is number of questions for the given task

Figure 6.2 shows a comparison between the quality of the answers from the two gropus of users for the *browsing* task. The X axis represents the questions, the Y axis indicates a quality value of the answer to the questions; this value is based on the *GroupQuality* mentioned above.

No substantial difference in results can be seen in the chart. For some questions (e.g. No. 1.4) Group 1 performs better, for others (e.g. No. 1.1) Group 2 is better. The *GlobalGroupQuality* of both groups is exactly the same (1.3). Moreover, these results represent a relatively high quality in the browsing task.

The main failures in accomplishing this task were caused by the fact that some subjects searched for a string which was exactly the same as the text used to indicate the desired topic. For instance, the string "training of researchers" (see Appendix B, task 1.2) does not appear in the collection; however there is a section entitled "training", which discussed just this topic. Obviously, this kind of problem depends on the indexing and searching system used, which is based on exact matching and does not include synonyms.

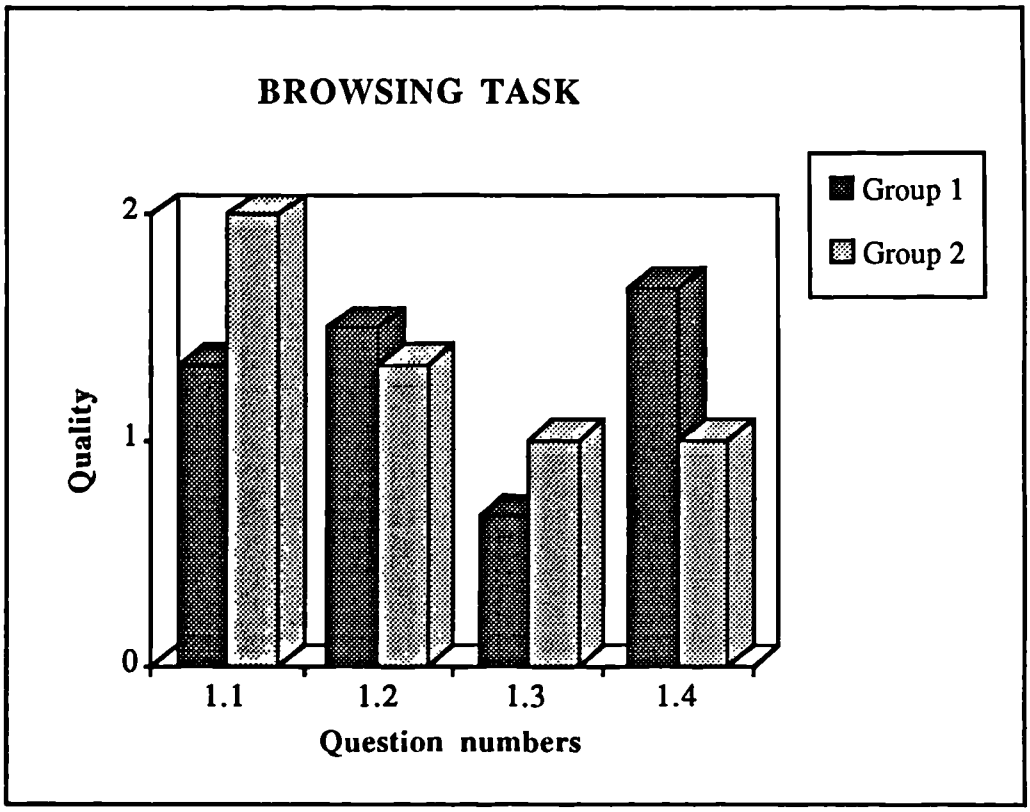


Figure 6.2: Answer quality in the browsing task

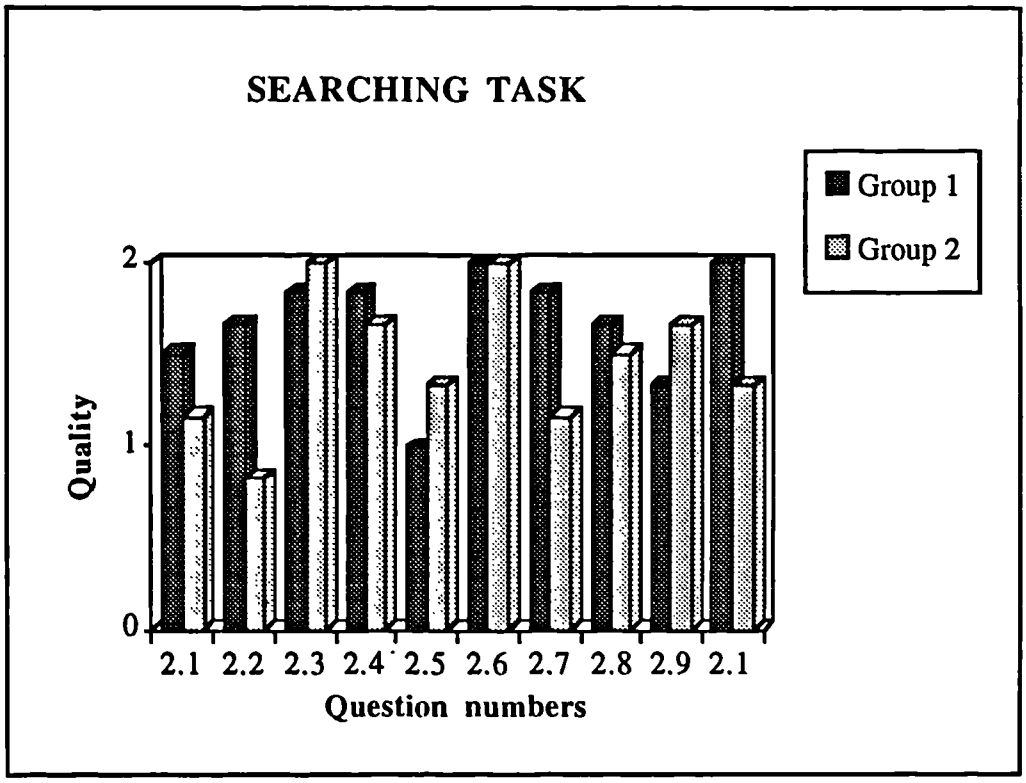


Figure 6.3: Answer quality in the searching task

Figure 6.3 shows a comparison between the quality of answers of the two groups of users for the *searching* task. The X axis represents the questions, the Y axis indicates a quality value of the answer to the questions; this value is based on the *GroupQuality* mentioned above.

Little difference can be seen in the chart. The *GlobalGroupQuality* of Group 1 is 1.7 while the *GlobalGroupQuality* of Group 2 is 1.5, on a scale from 0 to 2. Not much difference emerges between the two groups even for this task.

Moreover, these results represent a relatively high quality in the searching task.

For the *analysis* task a quality value is not significant because it is a very subjective task: for instance, it is difficult to judge that a certain criterion for highlighting text is better than another. However, analysis has been considered because it is a typical operation a user could perform on a book. What is more interesting for the purpose of this evaluation is to investigate which tools are mostly used during the task analysis. This will be presented in section 6.3.3.3.

In conclusion, for both the browsing and the searching tasks the results indicate a similar and good performance for both groups. In particular, the high percentage of correct answers underlines a correct and consistent displaying of information (book contents).

Another important factor to be considered, is the time taken by users to accomplish a task, which is the subject of the next section.

6.3.3.2 Time Taken to Accomplish Tasks

As mentioned above, the time measurement provides an indication of how easily the users found it to perform each task. An indication of the frequency of out of time answers has been already given in the previous section. For both the browsing and the searching tasks, the percentage of out of time answers is considerably low.

Time measurements have been derived from the log files. During the evaluation the subjects were invited to talk aloud. Sometimes, when the user had several comments, the time spent to execute a task increased; for these situations the evaluator took note of the time spent for commenting and this has been taken into account in the time computation.

As for the answer quality, a number of parameters have been considered in measuring time. The time taken by each user to answer a certain question is referred to as *UserTime*. This time has been directly derived from the log file of a each user.

$UserTime_{k,i}$ = the time taken to answer the question k by the user i

Successively, an average time in each group (*GroupTime*) was computed over the *UserTimes* in that group:

$$GroupTime_k = \frac{\sum_{i=1}^{N} UserTime_{k,i}}{N}$$ where N is number of subjects in the group

Both *UserTime and GroupTime* are relative to a single question. *GroupTimes* are the time values shown in the following charts. Another parameter which was interesting to consider is the *GlobalGroupTime*, which represents the average time required for a given group to answer the set of questions assigned for a certain task:

$$GlobalGroupTime = \frac{\sum_{k=1}^{M} GroupTime_k}{M}$$ where M is number of questions for the given task.

Figure 6.4 shows a comparison between the time taken by the two groups of users to accomplish the browsing task. The X axis represents the questions, the Y axis indicates the average time taken by each group to answer the questions (*GroupTime*). The chart indicates that little difference exists between the two groups. The *GlobalGroupTime* for the browsing task was between 2 and 3 minutes for both groups, although a slightly better performance was achieved by Group 1.

From the analysis of the log files and from the users' answers it emerges that all the users were able to answer most of the questions in the assigned time (four minutes). The average number of answers given in the maximum time by Group 1 was 3.6 out of 4 questions. The average number of answers given by Group 2 was 4 out of 4 questions; this means that all users answered all the questions in the assigned time.

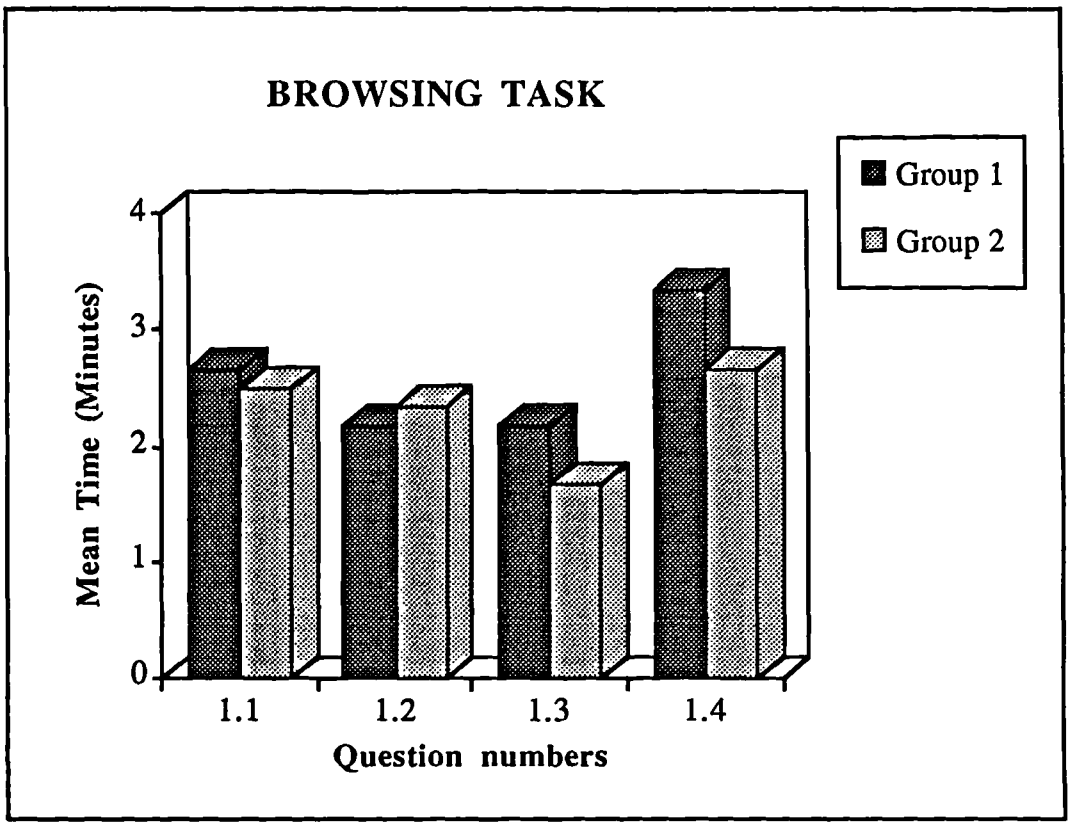


Figure 6.4: Time taken in the browsing task

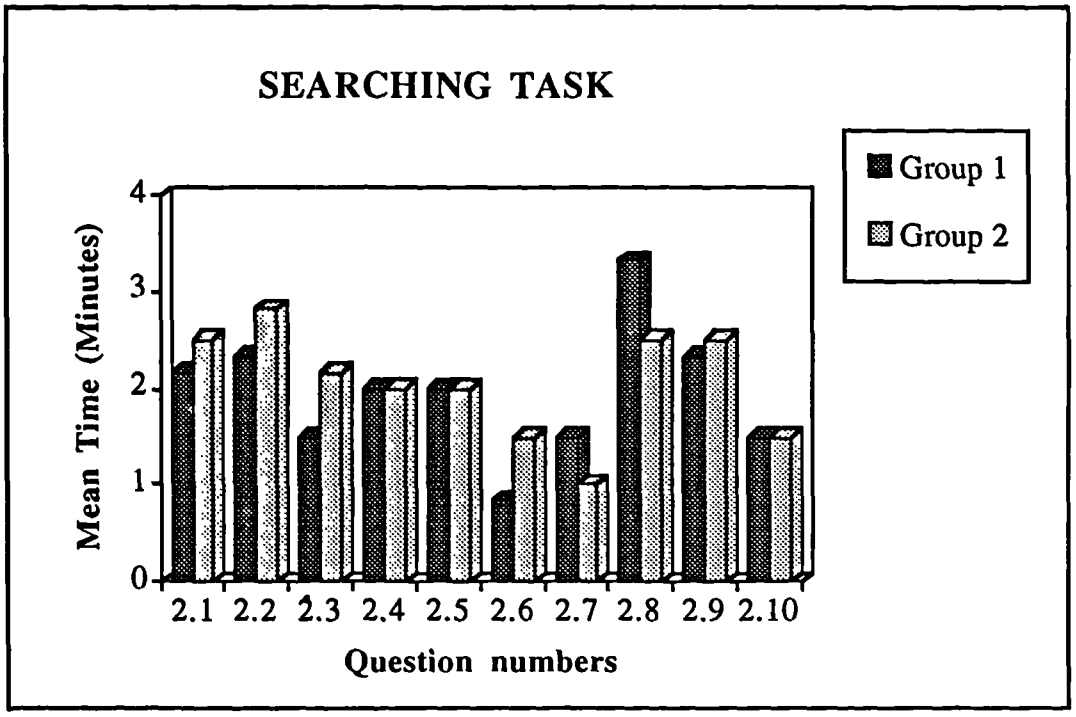


Figure 6.5: Time taken in the searching task

Figure 6.5 shows a comparison between the time taken by the two groups of users to accomplish *searching* task. The X axis represents the questions, the Y axis indicates the average time taken by each group to answer the questions (*GroupTime*).The chart indicates that little difference exists between the two groups. The *GlobalGroupTime* for the searching task was around 2 minutes for both groups.

From the analysis of the log files and from the users' answers it emerges that all the users were able to answer most of the questions in the assigned time, four minutes. The average number of answers given in the maximum time by Group 1 was 9.6 out of 10 questions. Exactly the same average number (9.6) of answers was given by Group 2.

For the *analysis* task, this kind of measurement does not make sense because a fixed amount of time, 10 minutes, was assigned for executing the task. This time was usually not sufficient for completely analysing a topic but it was enough to give indications of how a user would execute it.

6.3.3.3 Frequency of Tool Usage

As earlier mentioned in section 6.2.4.3, the frequency of tool usage is another interesting aspect that can be extrapolated from the data collected during the evaluation. This section indicates the frequency of usage of the tools (expressed as a percentage) while executing the browsing, searching and analysis tasks, in the two groups of users.

This can be useful in order to investigate the users' strategies while executing their tasks. These results are derived from the log files. As described above, any tool activated by the user was recorded in a log file. An automatic procedure assigns a group identifier to each tool (see below) and computes the frequency of tool usage.

The tools provided in the *hyper-book* system have been classified into several groups, according to their functionalities, as anticipated in section 4.4 of chapter 4. These categories are summarised below, indicating for each category a group identifier:

- P, personalisation (add/delete bookmark, add/delete notes, annotation);
- H, history (return, recent, bookmarks);
- HL, hierarchical links (from Table of Contents, from List of Tables, from List of Figures, from Index);

- TL, transverse links with the exception of external links (xi, g, acr, n, r, zoom);
- LA, linear access (next/previous page, flip pages);
- DIA, direct internal access (go to page, thickness, Table of Contents, List of Tables, List of Figures, Index, Front Cover, Back Cover);
- DEA, direct external access (bookshelf, report15, report17, xe);
- ES, external sources (print, copy text, word processor)
- S, search.

P, H, HL, etc. represent the identifiers of each group used in the log file.

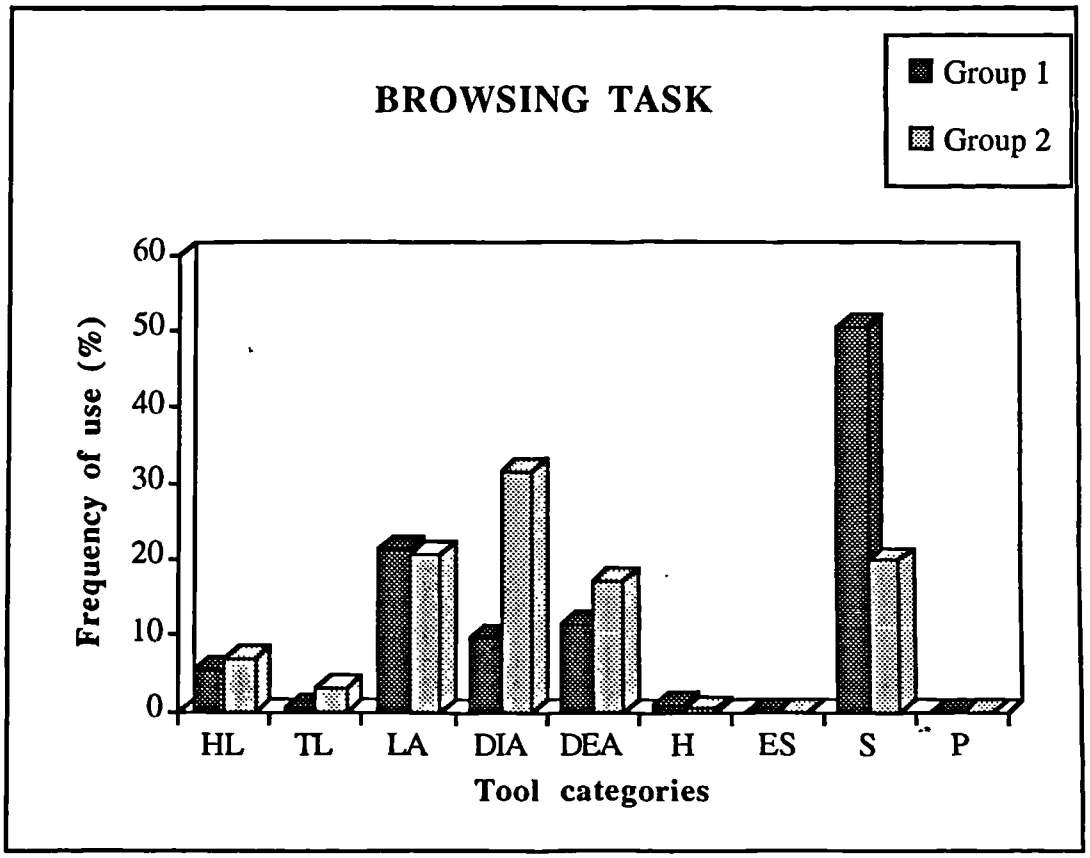


Figure 6.6: Frequency of tool usage in the browsing task

Obviously the kind of tools used depends on the type of task. For the browsing task, for instance, as indicated in figure 6.6, navigation and searching tools are the most used. A number of considerations emerged from the results shown in figure 6.6:

- computer experts used the search tool more than computer non-experts, who preferred direct internal access tools, mainly "Go to Table of Contents" and then activated Table of Contents entries (HL);
- transverse links were not used a lot in general; they were mainly used by computer non-experts;

- linear access was adopted to a noticeable degree by both groups;
- direct external access was quite used as the task requires browsing in the collection;
- history, personalisation and external sources were not used because the browsing task did not require them.

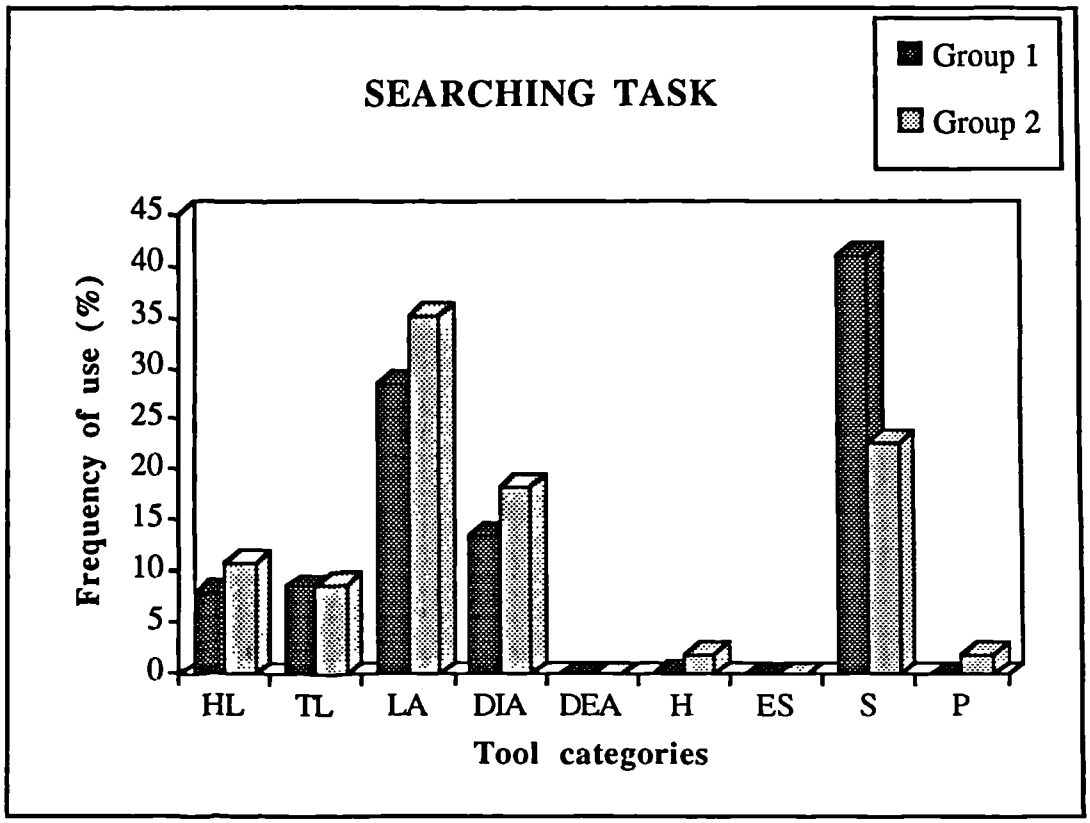


Figure 6.7: Frequency of tool usage in the searching task

Figure 6.7 shows the frequency of tool usage for the searching task. A number of considerations emerged from the results shown in figure 6.7:

- linear access tools were extensively used by both groups;
- searching tools were more widely used by computer experts, while computer non-experts preferred to browse through the Table of Contents (DIA) and activate Table of Contents entries (HL);
- transverse links were quite used by both groups;
- direct external access tools were not used as the task required the use of a single document;
- external sources tools were not used because the searching task did not require them;

- history and personalisation tools were occasionally used but just by computer non-experts.

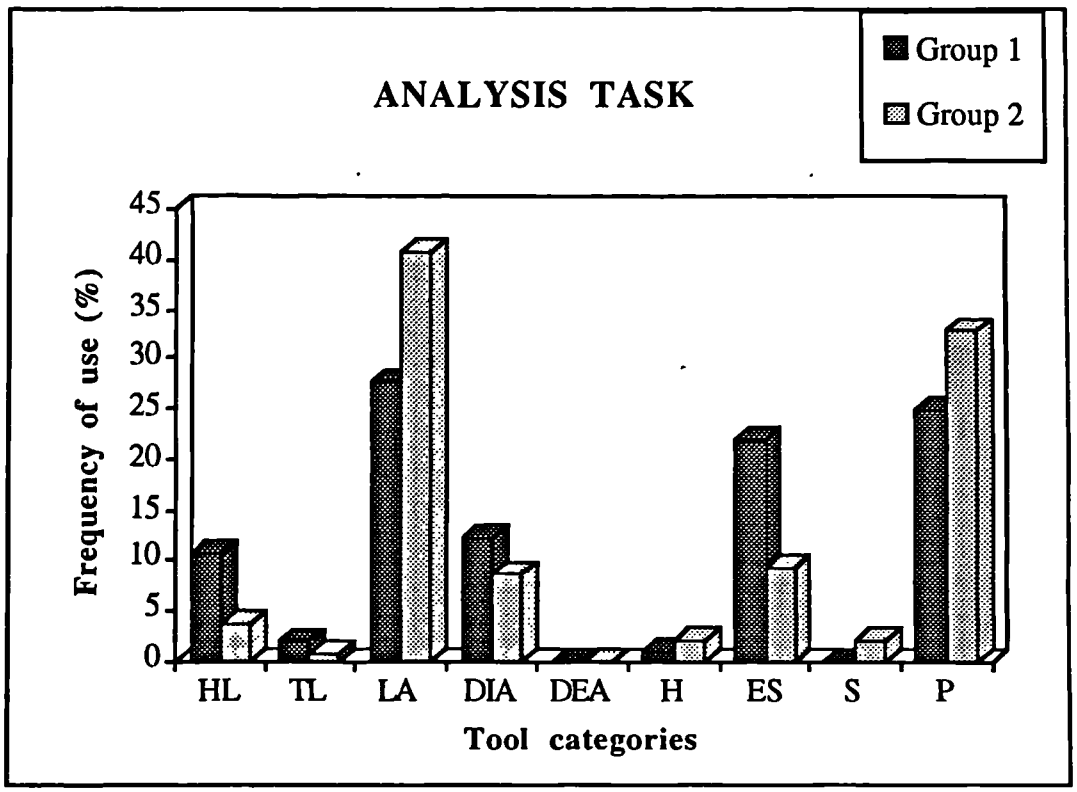


Figure 6.8: Frequency of tool usage in the analysis task

Figure 6.8 shows the frequency of tool usage for the *analysis* task. In the main, two different strategies have been used for the analysis task:

- some subjects preferred to use personalisation tools for highlighting the relevant parts in the document;
- others preferred to "export" all the relevant parts (copy and paste) and make their elaboration outside the document using a wordprocessor.

The first strategy (personalisation tools) was mostly adopted by computer non-experts, while computer experts used both strategies as indicated below:

Group 1 (experts)

use of word processor:	3 users
use of personalisation tools:	3 users

Group 2 (non-experts)

use of word processor:	1 user
use of personalisation tools:	4 users
use of word processor + personalisation:	1 user

This result was not unexpected as people who have a limited experience with computers employ tools which they normally use with paper documents, while computer experts are more aware of the capabilities of the computer and are already used to employ them. Another possibility, which nobody adopted, would have been to print the relevant parts and treat them as in the paper version.

By analysing the chart it emerges that:

- direct internal access and hierarchical links were adopted by both groups (they were used mainly at the beginning in order to identify the topic). At this stage of the evaluation, users already knew the document, therefore, it was not difficult for them to find out where the topic was discussed in the document;
- linear access tools were widely used: once the place where the topic started was located, users browsed forward and backward in order to identify all relevant pages;
- personalisation tools were widely adopted, mainly by computer non-experts;
- external sources were widely adopted, mainly by computer non-experts;
- history and transverse links were marginally used;
- external direct access tools were not used at all, because the analysis task did not require them;

6.3.4 Results from the Final Questionnaire

These results have been derived from the final questionnaire which contains questions about the system usability (learnability, errors and user satisfaction) and about the system utility.

6.3.4.1 System Usability

Ease of learning

First of all, users were asked whether they found the system easy to learn. The answer could range from 1 (very easy) to 5 (very difficult). The average of the answer values given by the users in each group was computed (*groupaverage*). The same procedure has been followed for all answers to the questionnaire.

Both groups judged the global system between easy and very easy to learn. The *groupaverage* of Group 1 was 1.50, while the *groupaverage* of Group 2 was 1.17.

Another question about learnability aimed at assessing users' opinions about single tools rather than the global system. The users were asked whether they found any particular tool difficult to learn or understand. Figure 6.9 reports the results indicating, for a given tool, the *groupaverage* of the two groups.

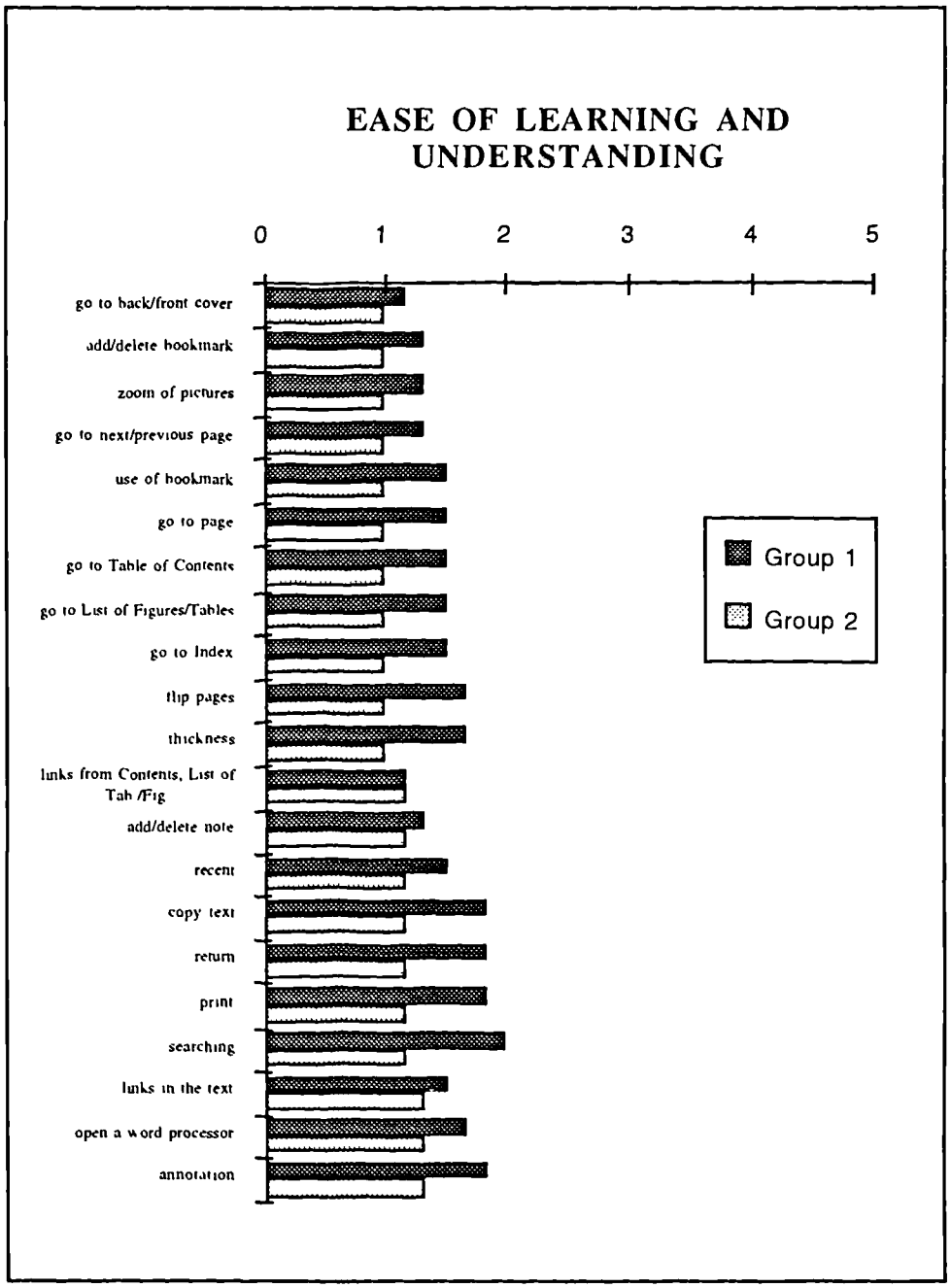


Figure 6.9: Ease of learning and understanding

All the tools received a high score: in fact, they were evaluated between easy and very easy to learn and understand by both groups on a scale from very easy (1) to very difficult (5).

For Group 1 tools can be ranked according to the assigned score as follows in an increasing order of difficulty:

- links from Table of Contents/ List of Tables/List of Figures - go to back/front cover	1.17
- add/delete bookmark - add/delete note - go to next/previous page - zoom of pictures	1.33
- recent - use of bookmarks - links in the text - go to page - go to Table of Contents - go to List of Figures/Tables - go to Index	1.50
- flip pages - thickness - open a word processor	1.67
- annotation - copy text - return - print	1.83
- searching	2.00

For Group 2 tools can be ranked according to the assigned score as follows:

- add/delete bookmark - use of bookmarks - go to next/previous page - zoom of pictures - flip pages - go to page - thickness - go to Table of Contents - go to List of Figures/Tables - go to Index - go to back/front cover	1.00
- add/delete note - copy text - return - recent - links from Table of Contents/ List of Tables/List of Figures - print - searching	1.17
- annotation - links in the text - open a word processor	1.33

The scores of computer non-experts range from 1 to 1.33 and most tools were assessed as very easy (1), whereas the scores of computer experts range from 1.17 to 2. This result could be explained by the fact that computer experts have more knowledge and expertise to judge the system tools.

Even if some tools were assessed easier to learn and understand than others, and computer non-experts assessed the tools as being slightly easier than did computer experts, there is not a qualitatively significant difference between them.

This result does not meet the expectation that computer non-experts had more difficulty in understanding those tools which do not have a correspondence in the paper book (e.g. searching, links, history, print, zoom pictures). This could probably be attributed to the fact that most subjects of Group 2 were already familiar with some of them (e.g. print) or that these tools were very simple to understand and use.

The objective of this question was to identify tools which were not easy to learn and understand. The reality was that all tools were assessed as being easy to learn and understand.

Error frequency

Another question was aimed at identifying if and where recurring errors occurred. As described in section 6.2.4.4, this refers to the errors a user commits because of misunderstanding the system tools. Most of the users in both groups (83% of users in Group 2 and 67% of Group 2) did not identify any errors. Among the others, some users were confused by some icons in the reader tool area (Contents, List of Figures, List of Tables). Others were confused because there was a small delay between their click on a link source and the actual activation of the link, and, consequently, they clicked more times on the source. Other users assumed that the selection of a portion of text automatically includes the copy operation; others tried to use navigation tools (e.g. go to next page) while an annotation palette was active.

In any case, users found it easy to correct their mistakes, whenever they occurred, showing a good fault tolerance of the system.

Another question dealt with whether the way of presenting information misled users. If the answer was yes, the user was required to indicate where the misleading source was. Most of the users in both groups (83% of users in Group 2 and 67% of Group 2) answered no. Among the others, some users were confused by the label

"return" of the return button. Others indicated as a cause of an ambiguity the content of the book. For instance, if the title of a chapter or of a section does not express clearly its contents, this could result in errors while accomplishing a task.

During the evaluation some bugs or weaknesses of the system were discovered. For instance the searching function did not indicate when a searching cycle was completed and the whole document had been scanned. Some people indicated them as recurring errors even if the purpose of the question was to identify user misunderstandings of the system functionalities and interface, rather than system bugs.

User satisfaction

Users generally expressed a good attitude towards the system. They were asked if they enjoyed using the system. On a scale from 1 (= enjoyed it a lot) to 5 (= found it very frustrating), the *groupaverage* of Group 1 was 1.83 and the *groupaverage* of Group 2 was 1.17. A slightly better attitude was expressed by Group 2 (computer non-experts).

Users were also asked whether they enjoyed having a book presented in this way. The answer was similar: Group 2 expressed a slightly better opinion (1.50) than Group 1 (1.67) on the same scale.

These results could be attributed to the fact that computer experts are usually more critical because they are aware of the computer capabilities.

6.3.4.2 System Utility

Missing Functions

In order to improve the prototype, the users were asked whether all the functions they liked to use were supported and, in case of a negative answer, which functions needed to be added or extended.

The majority of users (83% of Group 1 and 67% of Group 2) suggested a number of improvements, which deal with:

- the search function: several improvements were indicated such as Boolean searching, searching backward and forward, better highlighting of the searched terms in the text, and partial matching;

- the copy function: users suggested to extend it in order to copy across pages and to include images;
- the need for short commands;
- the need for parallel access to several books and inside the same book to more pages at the same time;
- the need for more meaningful bookmarks: they could be improved, for instance, by associating a significant name (as indicated in the design chapter);

Other improvements were identified only by Group 1. They can be summarised as follows:

- more flexibility for some aspects of the system: for instance reader tools should be available in a palette, reader tools should be more adaptive to the context, more synthetic dialogues are required in some cases (e.g. print, go to page), more responses should be given by the system (e.g. change of the cursor when it is over an active area);
- the possibility of defining new links;
- the possibility of browsing directly to the next/previous chapter;
- the need for online help.

Group 2 identified other limitations and suggested some improvements:

- the need for the number of pages in the Table of Contents: the number of pages behind each entry is considered as an important clue in order to evaluate how long a certain chapter is;
- the need for larger miniatures of the pages in the recent function: the actual size does not allow them to be recognised;
- the need for a button for directly accessing the references;
- the need for keeping the annotation palette active, while navigating the book.

A general result is that computer experts were more critical and gave more suggestions about the possible ways for improving the system, although the main problems were identified by both groups.

Tools Utility

Users were also asked to judge the utility of each of the tools provided in the *hyper-book* system. The answer could range on a scale from 1 (very useful) to 5 (not useful

at all). All the tools received a high score by both groups, although there are some differences. Figure 6.10 reports the results indicating, for a given tool, the *groupaverage* of the two groups.

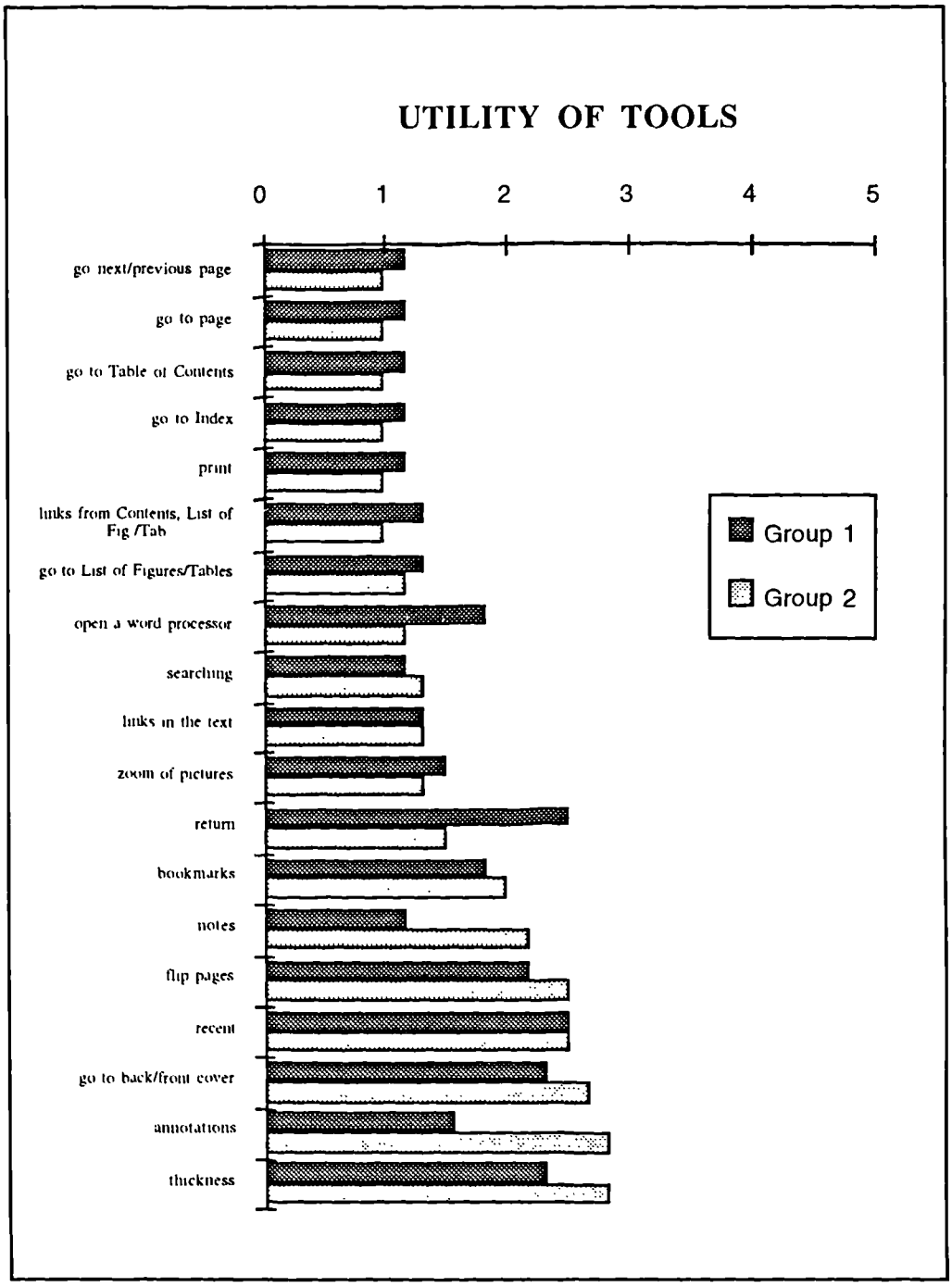


Figure 6.10: Utility of tools

For Group 1 tools can be ranked according to the assigned score in decreasing order of utility as follows:

- notes - go to next/previous page - go to page - go to Table of Contents - go to Index - print - searching	1.17
- links from Table of Contents/ List of Tables/List of Figures - links in the text - go to the list of Figures/Tables	1.30
- zoom of pictures	1.50
- annotations	1.58
- bookmarks - open a word processor	1.83
- flip pages	2.17
- go to back/front cover - thickness	2.33
- recent - return	2.50

For Group 2 tools can be ranked according to the assigned score as follows:

- links from Table of Contents/ List of Tables/List of Figures - go to next/previous page - go to page - go the Table of Contents - go to Index - print	1.00
- go to the list of Figures/Tables - open a word processor	1.17
- links in the text - zoom of pictures - searching	1.33
- return	1.50
- bookmarks	2.00
- notes	2.17
- recent - flip pages	2.50
- go to back/front cover	2.67
- annotation - thickness	2.83

In general, the two groups agreed in assessing the following tools as the most useful: direct access to Table of Contents, List of Tables/Figures, Index, go to next/previous page, direct access to a specified page, links, print, open a word processor, searching and bookmarks. Other tools such as recent, flip pages, thickness, go to back/front cover have emerged as being less useful. In some cases, such as recent and flip pages, the reason was attributed to the way the function was implemented rather than to the intrinsic nature of the function itself. For instance, the slowness of flipping pages makes response time unacceptable; the size of the page miniatures in the recent function makes pages unrecognisable.

The go to back/front cover and the thickness tools were also assessed as not being very useful. There are two possible explanations:

- users worked with a small collection of books during the evaluation, whereas the back/front covers and the thickness could be more useful in a large collection. In this case the users' experience during the evaluation session could have influenced their assessment;
- another possibility is that they might not be necessary in general. This could lead to the conclusion that there is no need to apply the book metaphor to this extent.

Personalisation tools (annotations and notes) did not receive a very high score by Group 2. This appears to contradict the results of the evaluation which indicated a large use of personalisation tools by Group 2. However, the extensive use of personalisation tools was limited to the analysis task, whereas the question here was about the utility of the tools in general.

Overall Utility

The users were also asked to express their opinion about the utility of the proposed approach for presenting electronic text.

Almost all subjects expressed a positive opinion towards this kind of text presentation (83% of Group 1 and 100% of Group 2) and gave a number of reasons for that:

- the system is easy and quick to use;
- navigation and searching capabilities provide a quick access;
- it is possible to personalise the book as well as to export portions of text and save excerpts;
- active links are available in the text;

- the ease of interchanging electronic documents makes things more efficient.

The remaining subjects (17% of Group 1) considered the system useful only if there is a collective effort to convert all paper books and documents into electronic form.

Another quite similar question was whether and in which situations users would use an electronic book. The answer was very similar to the answer given in the previous question. The percentage of subjects who were completely favourable was over 67% of Group 1 and 100% of Group 2. The other 33% of Group 1 answered positively in some situations and negatively in others. The two groups gave similar answers; therefore no distinction is made in the following.

Users indicated a number of typical situations during which they would use an electronic book:

- searching or navigating;
- studying the book;
- using the book as a reference material.

Sequential reading, on the other hand, was a typical situation in which the electronic book was not considered very useful. For sequential reading users expressed a strong preference for the paper book. If the paper version did not exist, readers preferred to use the print tool to generate one rather than read text on the screen. The electronic version, although very similar to the paper version does not guarantee the same comfort (low screen resolution, difficult portability).

6.4 Conclusion

This session discusses the results of the evaluation and draws some conclusions. The main findings have been derived from the evaluation session (which have provided measurements of the quality of the answers, the time taken to accomplish tasks, and the frequency of tool usage) and from the final questionnaire (user's opinion about the system usability and utility).

The participation of experts (both in the area of the test application,*Global Change* , and in the evaluation area) in the preparation and execution of the evaluation gives credit to the evaluation both for what concerns the choice of suitable tasks and the adopted procedure.

Results from the Evaluation Session

Some conclusions can be derived from the evaluation session. Both groups achieved a good performance and no evident difference in quality emerged for the browsing and searching tasks.

A similar result was obtained from the time measurement: no significant difference emerged between the browsing and searching tasks, and the large majority of subjects could answer the questions in the assigned time.

Therefore, computer expertise does not represent a significant aid in order to perform better or execute tasks quickly. These results lead to the conclusion that the system is easy to learn and use by everyone, and supports the initial hypothesis.

Other interesting results arose from the analysis of the frequency of tool usage. This allows the users' strategy in executing a given task to be elicited, and, in some way, provides an indication of those tools which are more useful for a certain task. The choice of the tools depends on two factors: the nature of the task and the experience of the user. For example, concerning the browsing and searching tasks, navigation (linear, hierarchical, direct access, etc.) and searching tools were the most frequently used by both groups because the tasks required these kinds of functionalities rather than personalisation tools or external sources (e.g. printer). However, computer experts preferred to use searching rather than navigation tools for the browsing and searching tasks because they were already familiar with them; on the other hand, computer non-experts tended to use a strategy (navigation) closer to the strategy adopted with paper books.

A similar result can be drawn from the analysis task. A significant number of computer experts preferred to export the relevant text in a word processor and then work on it separately. This choice can be attributed to their knowledge of computer programs. On the other hand, computer non-experts preferred to employ personalisation tools, using a strategy which depended on their experience with paper books.

In general, people tended to employ those tools which were more obvious and self-evident according to their previous experience. The *hyper-book* system provides tools in order to meet different needs and habits.

Another interesting result is that linear access is the tool more widely used by both groups in all the tasks. The possibility of moving forward and backward page by page in the document allows users to get rapidly familiar with or recognise a certain part of text.

There are some tools (mainly history tools) which were marginally used by both groups. Two possible conclusions can be drawn: one reason may be that they were not required by the tasks assigned; a second may be that they were not considered useful in general. The results from the final questionnaire seem to indicate that they are not judged very useful in general.

Results from the Final Questionnaire

Other conclusions can be drawn from the sets of results of the final questionnaire. They are connected with the findings reported above, but, in this case, the results were obtained by asking users directly rather than extracting them from the execution of tasks.

The users' opinion about the ease of learning confirms the result already reported. In fact, the global system as well as the specific tools were assessed between easy and very easy. The answers to the questions about users' errors indicate that they were rarely misled by the system and few errors were committed. The system was found to be simple and clear, and generally satisfied users.

Other questions aimed at identifying the missing or weak functions and the utility of the overall system as well as of the specific tools. Both groups expressed a similar opinion even if computer experts were more critical and observed more limitations on the basis of their knowledge of computer capabilities. According to the users' opinion, a number of functions need to be extended (e.g. searching, copy) and some new functions have to be added (e.g. short commands, parallel access to more books and to more pages inside the same book, the number of pages in the Table of Contents, etc.).

In general users recognised a high utility of the global system and appreciated the approach adopted for presenting electronic text because it is easy and quick to use, it can be easily accessed by means of the navigation and searching tools, and it can be easily personalised. Moreover, the system was recognised to be very useful for consulting a book, searching for a specific item of information, and as a reference source. On the other hand, it was not considered useful for sequential reading, for which users expressed a strong preference for the paper book.

Another question was used to identify the level of utility attributed to the single tools. Most of the tools were assessed as being quite useful; in particular direct access to Table of Contents and Index, links, print, searching, go to next/previous page and direct access to a specified page were assessed as the most useful tools by both groups. On other hand, other tools such as recent, flip pages, thickness, go to

back/front cover have emerged as being less useful. For some of them, such as recent and flip pages, this probably depends on implementation limitations and not on the intrinsic nature of the function itself.

In conclusion, users' opinion about the system usability and utility has permitted the identification of a number of improvements which will be valuable and will be indispensable with respect to enhancing the current version of the prototype in order to produce a more acceptable system.

In spite of the difficulties and problems which have been indicated, the overall result shows a positive response and attitude from different categories of users towards the chosen approach. Therefore, the *hyper-book* model can be considered sufficiently adequate to present electronic texts and helpful for a wide range of users.

Chapter 7:

Conclusions and Future Developments

The central objective of this research, as introduced in chapter 1, has been to demonstrate that it is appropriate to translate the metaphor of conventional paper books into an electronic environment, by reproducing and extending features and functionalities of paper books in the computerised form. In chapter 2 the literature background for this research was provided. The implications from the literature review (chapter 3) have led to a more precise definition of the context, the publishing process, and to a clear focus of research, which involves the acquisition and authoring processes of the author's manuscript in order to produce an electronic document provided with appropriate presentation and readers' services. A formal model of an electronic book *(hyper-book)* and an environment for producing electronic books (the *hyper-book builder*) have been defined (chapter 4) and successively implemented (chapter 5). A prototype application was developed in order to evaluate the *hyper-book* model with a potential group of users (chapter 6). The results of the evaluation have been compared to the expectations expressed by the initial hypothesis and have provided a number of suggestions for possible improvements and extensions of the prototype.

The conclusions and the lessons gained from the work carried are summarised below.

7.1 Conclusions

In the light of the work carried out and described in the previous chapters, a number of conclusions can be drawn.

General conclusions

- *Have the objectives been achieved?* The objective of the thesis has been reached. What was proposed at the beginning of this research has been demonstrated; namely, the appropriateness and possibility of translating the book metaphor into an electronic environment. This has been realised through the definition of the *hyper-book*, which represents a model for an electronic book, and the implementation and evaluation of this model.

- *Tangible Results.* The implementation of a prototype, which includes both the *hyper-book* and the *hyper-book builder*, has permitted the ideas and concepts expressed in the theoretical model to be demonstrated. In addition, it has provided the environment within which to carry out an empirical evaluation.

- *Difficulties.* Some difficulties were encountered both in finding the source texts for *hyper-book builder* and in defining an evaluation strategy for *hyper-books.*

Concerning the acquisition of the source texts, the initial idea was to use documents already marked in SGML. A requirement was that the DTD had to be appropriate for describing the structure of a scientific book and, at the same time, not excessively complex as the prototype had to demonstrate concepts rather then to be used for commercial purposes. In addition, the documents had to be chosen so that it was quite easy to find potential users for the evaluation phase. Several attempts to find documents which satisfy these requirements, by contacting publishing houses and other organisations, did not succeed. Therefore, the adopted solution was to define first an appropriate DTD (the *hyper-book DTD*), based on an existing DTD, and then mark up existing ASCII texts according to this DTD. The choice of the texts was determined by the general project within which this research has been conducted.

Another difficulty dealt with the definition of an evaluation strategy for *hyper-books.* It was hard to find reference literature specifically about the evaluation of this type of electronic book, because it represents a quite innovative approach for presenting electronic texts. Therefore, the choice was to search for material about

evaluation in general and to look for examples of evaluation of specific systems which are similar to some extent to the *hyper-book* (e.g. hypertext and other kinds of electronic text representation).

Based on my experiences in developing and evaluating the prototype, several results concerning architectural features are worth noting. In the following a distinction is made between the *hyper-book* and the *hyper-book builder.*

Conclusions on the hyper-book

Some positive results related to the *hyper-book* have been drawn from the evaluation.

- *Effectiveness of the book metaphor.* This is the most important result of this research. The evaluation has demonstrated that the choice of maintaining the book metaphor really guarantees ease of learning and using for everyone. A minimum training period was sufficient to get users familiar with the system. Computer expertise does not represent a significant advantage in using the *hyper-book* system better. In fact, the two groups of people chosen for the evaluation, computer experts and computer non-experts, both achieved a good performance in executing the tasks assigned and agreed that the system and the reader tools were easy to learn and use. Computer knowledge represents an important factor in determining the strategy used for accomplishing a given task: people tend to use tools which are related to their previous experiences.

- *Utility of the hyper-book system.* Users recognised a high utility of the overall system and indicated some situations in which the system could be particularly useful. In addition they attributed a high level of utility to most of the reader tools provided.

The *hyper-book* system has been considered useful mainly for consultation, searching, and as a reference source, thanks to the navigation, searching and personalisation functions. However, the paper book has been preferred to the electronic book for some tasks such as sequential reading.

The tools which have been assessed as the most useful by both groups of users include: direct access to the Table of Contents, activation of the Table of Contents entries, searching, print and linear access (go to the next and previous page). Other tools were assessed as being less useful: recent, flip pages and thickness.

Despite the successful performance of the users and their positive attitude towards the system, they have indicated some weaknesses in various aspects. It is important to underline that the prototype implements most of the features of the theoretical model. However implementation issues have imposed some constraints on the original model. Therefore, most of the limitations identified by the users are dependent on practical issues rather than on design choices.

Design limitations include:

- *Static links structure*. Readers cannot modify and extend the existing link structure. It could be very useful, mainly in the context of an electronic library, to provide the readers with the possibility of defining their own paths across the library, so that a connection between relevant material can be defined and kept.

- *Lack of page numbers in the Table of Contents.* In a paper book, the role of the page number beside each Table of Contents entry is mainly to provide the address of the location of chapters, sections and subsections in the book. In an electronic book, active Table of Contents entries allow the reader to access directly a section with a simple click without the need for flipping through the book until the right page is found. However, another important role has been recognised by most users: page numbers are an important clue in order to know, for instance, the length of a certain chapter directly from the Table of Contents. This possibility is currently missing in the *hyper-book* .

- *More flexibility*. This concerns, for instance, the fact that:
  - reader tools are not adaptive to the context,
  - parallel access to more books and to more pages is not allowed,
  - short commands are not provided;
  - bookmarks need to be better specified: a page number is not a meaningful label.

Limitations which are more dependent on implementation issues include:

- *The search function*. The search function exploits the built-in HyperCard *find* function. This has avoided the development of a new indexing and searching tool; however it has some limitations according to the design choices: no feedback on the Table of Contents is visible after a search, and search terms are highlighted in the text

one at a time. In addition, no Boolean search or proximity search is provided, no indication when a search cycle has completed is given, etc.

- *The copy function.* The copy function does not allow pages to be copied across and figures and tables embedded in the text to be copied together with the text. The reason is that it is not possible with the current HyperCard version to select text from different HyperCard fields at the same time. In addition, figures and tables are not really parts of the HyperCard fields, but have been overlaid by means of external commands.

- *Inefficiency.* This involves mainly the access to pages which contain graphics, the use of personalisation tools such as annotation, and the flip pages function.

- *Low utility of the recent function as history tool.* Low utility has been attributed to the recent function, because the page miniatures are too small in order to be recognisable.

- *Personalisation tools.* Some personalisation tools need to be extended according to the design choices. For instance a highlighter is not currently available; a note manager could be added in order to allow the reader to retrieve and organise his/her own notes.

Conclusions on the hyper-book builder

It is possible to identify a number of positive aspects and results related to the *hyper-book builder.*

- *Traditional Publishing Systems.* The *hyper-book builder* presents several features which make it closer to a traditional publishing environment rather than to existing so-called electronic publishing systems (e.g. FrameMaker, Ventura Publisher, Quark XPress). The main difference between the *hyper-book builder* and existing electronic publishing systems lies in the fact that the former maintains the distinction between the traditional roles of author, publisher and reader, while the later usually does not consider such a distinction. The *hyper-book builder* is a system conceived for a publisher who receives the original manuscript by the author (who provides the content and the logical structure of the text) and produces the final electronic book by associating a layout style and some reader services. Existing

electronic publishing systems are usually packages for the preparation of electronic documents rather than systems for publishing in the sense of selling in the public domain. The author produces a document, by combining text, images and other multimedia components, and defines the logical structure and the layout style of the document. In summary, here the author also incorporates the role of the publisher.

- *Saving production costs.* The broad potential of the *hyper-book builder* deals with saving production costs of electronic books. Most of the systems which present electronic text or multimedia information have been produced ad-hoc for a specific application. The *hyper-book builder* has been designed in order to automate the publishing process, once the original text is available in SGML. The publisher, having defined the layout style and the reader services in the empty template, can automatically produce as many electronic books as required, thereby saving considerable production costs.

- *Typographical Flexibility.* The publisher can define and change the book size, page size, typefaces, and other typographical aspects of the *hyper-book*. This guarantees that the system has good adaptability when different screen sizes and resolutions are used. For example, the same book could be presented on a smaller screen by reducing the book size. From a practical point of view, the current prototype does permit the book size and other typographical features to be changed but does not allow the area occupied by the editor and reader services to be reduce or re-organised. This limitation could be overcome by using a palette for the reader services and for the editor tools.

- *Reader Services Flexibility.* The possibility of managing the reader services allows the publisher to extend the set of tools provided to the reader with other tools which may be appropriate for a book in a specific domain.

Despite these positive aspects, some weaknesses and limitations have been identified. These contribute, together with the limitations indicated for the *hyper-book* model, to set the agenda for further work and development.

Most of the problems which have been identified depend on implementation issues, others are intrinsic in the design choices.

Problems which depend on design issues include:

- A need to provide texts marked up according to the *hyper-book DTD* as sources. Although SGML has a number of advantages as described in chapter 2 (e.g. it guarantees independence both from the system and the application), there is a degree of uncertainty that authors would agree to submit their manuscript in SGML. In fact, this is a very hard job if the process is carried out manually, i.e. without a proper SGML editor (as the experience of this work can confirm). On the other hand, even if SGML editors can facilitate the mark up activity, some knowledge of the basic SGML philosophy (e.g. the concept of tags) is still required. A solution could be to develop more transparent editors or use an intermediary between author and publisher.

Problems, which are dependent on implementation issues (see chapter 5) include:

- *Limited text formatting capabilities.* Text cannot be justified, a built-in pagination function is not provided, limited typographical capabilities are available. Some of these deficiencies have been solved by implementing the missing capability; for instance a pagination tool has been developed. Others, e.g. justification, will be solved by using future versions of HyperCard.

- *Inefficiency.* This is particularly evident during highly computational operations such as importing and repaginating text. The slowness depends on the fact that HyperTalk, the HyperCard script language, is an interpreted language.

- *Limited image management capabilities.* HyperCard does not provide advanced tools for managing figures and tables, except by using external commands. The result is that images and tables are not well integrated into the HyperCard system.

7.2 Future Developments

Here possible extensions, improvements and elaborations to the work reported in the thesis are described. They concern extensions of the system within the more general context of the electronic publishing process, improvements of the *hyper-book*, and improvements of the *hyper-book builder*.

Among the possible extensions of the system within the general context of electronic publishing proposed in chapter 3, the most important are:

- *The realisation of a full-text electronic library.* This is the most immediate extension of the *hyper-book* system: *hyper-books* have been designed in the context of an electronic library. Their potential within this context has been introduced in section 4.3. It mainly concerns:

  - the acquisition and processing of the material: the *hyper-book builder* incorporates the functions for automatically producing books which are suitable for an electronic library;

  - automatic cataloguing, indexing, and abstracting, which could be achieved by exploiting the documents' original markup;

  - the provision of an user-friendly interface for electronic books, which has been implemented in *hyper-books*;

  - the provision of inter-document links, which creates a network of interconnected documents;

  - the fact that *hyper-books* do not suffer from the constraints of the paper books; therefore the same book may be replicated as many times as required.

  In order to realise a full-text electronic library, other issues need to be considered; they include: intelligent tools for simulating the human reference librarian; loans management; book delivery; copyright; and search and retrieval management, taking into account gateways to other library systems.

- *Extension of the Book Metaphor.* The choice of maintaining a high degree of similarity with the paper book, in principle, aims to involve users as much as possible in the use of electronic books. Once people are used to such a system, it will be possible to move gradually towards more and more powerful, flexible and dynamic systems. The result of this evolution could go far from the initial model, and the book metaphor could be partially or completely abandoned in future. Therefore the adoption of the book metaphor has to be considered as a starting point rather than as an inflexible paradigm.

- *Inclusion of new types of book.* Two issues have to be considered when designing electronic books: the nature of the embedded information, and the types of service they have to provide. These two issues have been used as criteria for classifying electronic books in several categories (Barker 1993), as mentioned in section 2.3.2: text books, static picture books, moving picture books, talking books, multimedia books, intelligent electronic books, and telemedia books, etc. *Hyper-*

*book*s belong to the static picture book category. An interesting extension would be to consider the other categories of book mentioned above.

- *Inclusion of other kinds of publication.* This work focused on a specific kind of publication: electronic books. It could be interesting to carry out a similar study on other kinds of publication: for instance journals, reference books, manuals, etc. This would involve issues such as the definition of a DTD for describing the different document structure and the definition of appropriate presentation and reader services. It is expected that a significant number of the reader services provided by the *hyper-book* (personalisation, history, etc.) may have a quite large employment.

- *Other mark-up languages.* In the context of electronic publishing, this research concentrates on one possible channel of publication, where the input documents are supposed to be in SGML. As indicated in the previous section, this can be a limitation, as it is difficult for an author to submit his/her manuscript in SGML. An interesting extension would be to develop a publishing environment which can accept other mark-up languages for the source documents such as ODA, Hytime, etc. There are currently available translators which allow electronic texts marked in various formats to be converted into SGML. A preliminary translation could be accomplished on the source text in order to produce a text in a format acceptable to the current *hyper-book builder*, which would not be changed. A more radical possibility is to change the *hyper-book builder*, in order to be able to accept directly different marked texts as input. This implies the definition of different translation rules according to the markup of the input text and could require some extensions of the current *hyper-book* model. For instance, if Hytime is used as markup language, new tools need to be developed for managing video and animation in the *hyper-book* model.

- *Issues related to quality control.* An interesting subject of investigation deals with the implications of electronic publishing on activities such as editing and refereeing which are traditionally carried out in print with the purpose of publication quality control. As these processes concern the contents and the logical structure of a document, rather than its physical appearance, the original SGML texts provided by the authors could be used. Therefore the editorial activities can be significantly facilitated by the fact that the text can be exchanged in a standard format (SGML).

The limitations indicated during the evaluation for the *hyper-book* suggest a number of possible improvements. Problems which are dependent on implementation issues

(e.g. limitations of the copy function, of the recent function, of some personalisation tools, inefficiency, etc.) could be solved by developing appropriate external commands where it is possible, by waiting for new releases of the software (HyperCard) or by re-implementing the system in another development platform.

- *Dynamic links structure.* Two possibilities are available: first, it is possible to extend the link definition function, currently available as an editor tool, to the readers; the second possibility is to create a new tool which is simpler and does not require the user to know about the link types; the user should just specify the link destination. The links network created by a user should not affect the original copies of the books, but should be saved as the user's personal material as with annotations and other personal additions.

- *More flexibility.* This includes, according to the results of the evaluation:

  - to make reader tools more adaptive to the context, for instance by deactivating tools which are not selectable in a particular situation;
  - to permit parallel access to more books and to more pages, by using overlapped windows or by dividing the screen space into different areas according to the number of objects accessed at the same time;
  - to define short commands for the most frequent operations;
  - to specify bookmarks in a different way, for instance by associating a name, a note, etc.

  Other ways to increase flexibility could involve the extension to the reader of some tools which are currently available only to the publisher such as changing the typographical style of a page, redefine the page size, etc.

- *Search function.* The search function can be improved by following the guidelines provided by the users. This could be realised using a more advanced tool than the HyperCard find function (e.g. Hyper-KRS) or by implementing an appropriate tool.

The limitations indicated for the *hyper-book builder* suggest a number of possible improvements and further work:

- *Evaluation of hyper-book builder* While the *hyper-book* model has been implemented and evaluated using a group of potential users, the *hyper-book builder* has been implemented with the intention of demonstrating functionalities and prove

concepts rather than to meet the commercial expectations of a publisher in terms of performance. Therefore, an empirical evaluation has not been accomplished on the *hyper-book builder*. It could be interesting to involve real publishers as users in order to evaluate the system goodness and acceptability in a commercial environment.

- *Implementation issues*. Problems which are dependent on implementation limitations such as inefficiency, limited text formatting capabilities, etc. could be solved, as indicated above, by using new releases of HyperCard or by re-implementing the system in another developing environment. However, the main concepts and ideas have been demonstrated in the prototype and these kinds of limitation do not represent a real problem.

While there is considerable scope for extending systems like the *hyper-book* and the *hyper-book builder*, an important contribution has been given to the study of appropriate models for organising and presenting electronic books and a number of important elements have been added to the repertoire of tools and techniques for producing electronic books.

References

Alberico, R. and Micco, M. (1990) *Expert Systems for Reference and Information Retrieval.* Westport: Meckler

Arms, C. The Context for the Future, in, Arms, C. (ed). (1990) *Campus Strategies for Libraries and Electronic Information.* Bedford: Digital Press, pp.332-356

Arms, C. The Technological Context, in, Arms, C. (ed). (1990a) *Campus Strategies for Libraries and Electronic Information.* Bedford: Digital Press, pp.11-35

Arms, W.Y. and Michalak, T.J. Carnegie Mellon University (1990), in, Arms, C. (ed). *Campus Strategies for Libraries and Electronic Information.* Bedford: Digital Press, pp.243-273

Arnold, K., Collier, M., Ramsden, A. (1993) Elinor - The Electronic Library Project at De-Montfort-University. *Aslib-Proceedings* 45(1). pp.3-6

Barker, P. (1991) Interactive Electronic Books. *Interactive Multimedia* 2(1). pp.11-28

Barker, P. (1992) Electronic Books and Libraries of the Future. *The Electronic Library* 10(3) June. pp.139-149

Barker, P. (1993) *Exploring Hypermedia.* London: Kogan Page

Barker, P. and Giller, S. (1991) An Electronic Book for Early Learners. *Educational and Training Technology International* 28(4) November. pp.281-290

Barker, P. and Manji, K. (1991) Designing Electronic Books. *Educational and Training Technology International* 28(4) November. pp.373-280

Barker, P. and Giller, S. (1992) Electronic Books, in, Saunders, D. and Race, P. (eds). *Aspects of Educational Technology*, Volume XXV: Developing and Measuring Competence. London: Kogan Page

Benest, I. D., Morgan, G. and Smithurst, M. D. (1987) A Humanised Interface To An Electronic Library, in, Bullinger, H.-J. and Shackel, B. (eds) *Human-Computer Interaction - INTERACT'87.* North-Holland: Elsevier Science Publishers B.V., pp.905-910

Benest, I. D. and Duric', D. (1991) Some Design Issues in the Automated Office Metaphor, in, *Conference Proceedings of the European X Window System User Group, Guildford, UK*, Cambridge: European X User Group. pp.56-69

Bernstein, M. (1990) An Apprentice That Discovers Hypertext Links, in, Rizk A., Streitz N. and André, J. (eds) *Proceedings of the European Conference on Hypertext, Hypertext: Concepts, Systems and Applicatios, INRIA, France November 1990.* Cambridge: Cambridge University Press, pp.212-223

Benyon, D., Davies, G., Keller, L., Preece, J. and Rogers,Y. (1990) *A Guide to Usability - Usability Now!* Milton Keynes: The Open University

Blake, K. (1988) The Electronic Book. *Library Hi Tech* 6(1). pp.7-11

Bogaerts, W.F. and Agema, K.S. (1992) *Active Library on Corrosion* [Manual] Amsterdam: Elsevier Science Publisher B.V. in conjunction with National Association of Corrosion Engineers

Brooks, H. M. (1985) Expert systems in reference work, in, Gibb, F. (ed) *Expert Sytems in Libraries Proceedings of a Conference of the Library Association Information Technology Group and the Library and Information Research Group*, London: Taylor Graham. pp.36-49

Brown, P. J. (1988) Linking and Searching within Hypertext. *Electronic Publishing* 1(1) April. pp.45-53

Brown, H. (1989) Standards for Structured Documents. *The Computer Journal* 32(6). pp.505-514

Bryan, M. (1988) *SGML An Author's Guide to the Standard Generalized Markup Language.* London: Addison-Wesley

Burrill, V. A. and Ogden, J. A. (1989) VORTEXT: The hard-backed Screen. *Electronic Publishing* 2(3). pp.143-156

Callahan, D.R. (1991) The Librarian as Change Agent in the Diffusion of Technological Innovation. *The Electronic Library* 9(1). pp.13-15

Carroll, J.M., Mack, R.L. and Kellogg, W.A. (1988) Interface Metaphors and User Interface Design, in, Helander, M. (ed). *Handbook of Human-Computer Interaction.* North Holland: Elsevier Publishers B. V., pp.67-85

Carroll, J. M. and Mack, R. L. (1985) Metaphor, Computing Systems, and Active Learning, *Int. J. Man-Machine Studies* 22. pp.39-57

Carroll, J. M. and Olson, J. R. (1988) Mental Models in Human-Computer Interaction, in, Helander, M. (ed) *Handbook of Human-Computer Interaction.* North Holland: Elsevier Publishers B. V., pp.45-65

Casorso, T.M. (1992) NCSU Digitized Document Transmission Project: Improving Access to Agricultural Libraries. *The Electronic Library* 10(5) October. pp. 271-273

Catenazzi, N. and Argentesi, F. (1991) Super-Library: An Environment for the Conversion of Linear Text into Hypertext, in, *RIAO '91 Conference Proceedings: Intelligent Text and Image Handling, Barcelona Spain, 2-5 April 1991*, pp.45-66

CEC, DG XIII-b [nd] *SGML Briefing Note.* [Bruxelles: CEC], pp.1-16

Collis, B.A. (1991) The Evaluation of Electronic Books. *Educational and Training Technology International* 28(4) November. pp.355-363

Conklin, J. (1987) Hypertext: An Introduction and Survey. *IEEE Computer* 20(9) September. pp. 17-41

Cooke, P. and Williams, I. (1989) Design Issues in Large Hypertext Systems for Technical Documentation, in, McAleese R. (ed). *HYPERTEXT: theory into practice.* Oxford: Backwell Scientific Publications, pp.93-104

Cronk, R.D. (1993) *Unlocking Data's Content.* Byte 18(9) September. pp.111-120

Dautriat, H. (1990) *Il Questionario. Guida per la Preparazione e l'Impiego nelle Ricerche Sociali, di Psicologia Sociale e di Mercato.* Milano: Franco Angeli

DeRose, S.J. (1991) SGML and Hypertext: Course Notes, in, *Proceedings of the Third ACM Conference on Hypertext, San Antonio, Texas, 15-18 December 1991*, New York: ACM Press

Dodd, W.P. (1990) Convergent Publication, or the Hybrid Journal: Paper plus Telecommunications. *Electronic Publishing* 3(1) February. pp.47-59

Egan, D.E., Remde, J.R., Gomez, L.M., Landauer, T.K., Eberhardt, J. and Lochbaum, C.C. (1989) Formative Design-Evaluation of SuperBook. *ACM Transactions on Information Systems* 7(1) January. pp.30-57

Egan, D.E., Remde, J.R., Landauer, T.K., Lochbaum, C.C. and Gomez, L.M. (1989a) Behavioural Evaluation and Analysis of a Hypertext Browser, in, *Proceedings of ACM CHI'89 Conference Human Factors in Computing Systems, Austin, Texas, 30 April- 4 May*

Egan, D.E. , Remde, J.R., Landauer, T.K., Lochbaum, C.C. and Gomez, L.M. (1991) Acquiring Information in Books and SuperBooks. *Machine Mediated Learning* 3, pp.259-277

Egan, D.E., Lesk, M.E., Lesk, M.E., Lochbaum, C.C., Remde, J.R., Littman, M. and Landauer, T.K. (1991a) Hypertext For The Electronic Library? CORE Sample Results, in, *Proceedings of the Third ACM Conference on Hypertext, San Antonio, Texas, 15-18 December 1991*, New York: ACM Press. pp.299-312

Electronic Book Technologies. (1990) *Dynatext*, [available from Electronic Book Technologies, One Richmond Square, Providence, RI 02906]

Electronic Publishing Services (1992) *Network Publishing A Multi-Client Study.* Electronic Publishing Services, December

Evans, D.A., Ginther-Webster, K., Hart, M., Lefferts, R.G. and Monarch, I.A. (1991) Automating Indexing Using Selective NLP and 1st-order Thesauri, in, *RIAO '91 Conference Proceedings: Intelligent Text and Image Handling, Barcelona Spain, 2-5 April 1991*, Vol. 1, pp.624-644

Feeney M. SGML. (1989) *Library and Information Briefings* May. pp.99-115

Feldman T. (1990) *The Emergence of the Electronic Book.* London: British National Bibliography Research Fund; Publishers Association [British National Bibliography Research Fund Report 46]

Fox, E.A., Hix, D., Nowell, L.T., Brueni, D.J., Wake, W.C., Heath, L.S. and Rao, D. (1993) Users, User Interfaces, and Objects: Envision, a Digital Library. *Journal of the American Society for Information Science* 44(8) September. pp.480-491

Frei, H.P. and Stieger, D. (1992) Making Use of Hypertext Links when Retrieving Information, in, Lucarella, D., Nanard, J., Nanard, M. and Paolini, P. (eds). *Proceedings of the ACM Conference on Hypertext, Milano, Italy, 30 November - 4 December 1992.* New York: ACM. pp.102-111

Frisse, M. E. (1988) From Text to Hypertext. *Byte* 13(10) October. pp.247-253

Frisse M.E. (1988a) Searching for Information in a Hypertext Medical Handbook. *Communications of the ACM* 31(7) July. pp.880-886

Frisse, M. E. and Cousins, S.B. (1989) Information Retrieval From Hypertext: Update on the Dynamic Medical Handbook Project, in, *Proceedings of Hypertext '89, Pittsburgh PA, November 1989*, New York: ACM, pp.199-212

Frisse, M.E., Cousins, S.B. and Hassan, S. (1991) WALT: A Research Environment for Medical Hypertext, in, *Proceedings of the Third ACM Conference on Hypertext, San Antonio, Texas, 15-18 December 1991*, New York: ACM Press [Technical Briefing], pp.389-394

Furuta, R., Plaisant, C. and Shneiderman, B. (1989) A Spectrum of Automatic Hypertext Constructions. *Hypermedia* 1(2). London: Taylor Graham, pp.179-195

Gibb, F. (1993) Knowledge-based Indexing in SIMPR: Integration of Natural Language Processing and Principles of Subject Analysis in an Automated Indexing System. *Journal of Document and Text Management* 1(2), pp.113-125.

Gibb, F. and Smart, G. (1991) Knowledge-based Indexing: the View from SIMPR, in, MacDonald, C. and Weckert, J. (eds). *Libraries and Expert systems.* London: Taylor Graham, pp.38-48.

Glushko, R.J., Weaver, M.D., Coonan, T.A. and Lincoln, J. E. (1988) Hypertext Engineering: Practical methods for creating a compact disc encyclopedia, in, *Proceedings of the ACM Conference on Document Processing Systems, Santa Fe, NM, 5-9 December 1988*, ACM, pp.11-19

Glushko, R.J. (1989) Transforming Text into Hypertext for a Compact Disc Encyclopedia, in, *Proceedings of the ACM Conference on Computer-Human Interaction - CHI'89, May 1989*, ACM, pp.293-298

Gomoll, K. (1990) Some Techniques for Observing Users, in, Laurel-B (ed). *The art of human-computer interface design.* Reading: Addison-Wesley, pp.85-90

Gonzales Juarros, E. (1992) SuperLibrary. A Full-text Electronic Document Access and Retrieval System Based on ATK/AMS, in, Andrew Consortium (ed). *Andrew Technical Conference. Carnegie Mellon University.* pp.XI-1-XI-8

Gonzales Juarros, E. (1992a) SL's Graphical User Interface to a Multimedia Document Management System, in, *Mensch und Maschine-Informationelle Schnitt-stellen der Kommunikation. Proceedins des 3. Internationalen Symposiums fur Informationswissenschaft (ISI'92), Universitat des Saarlandes. Saarbrucken, 5-7 November 1992.* Universitatsverlag Konstanz, pp.86-93

Goodram, R.J. and May, J.H. (1991) The E-library: an Intelligent Environment for Expert Systems, in, McDonald, C. and Wecker, J. (eds) *Libraries and Expert Systems Proceedings of a Conference and Workshop at Charles Sturt University - Riverina, Australia, July 1990.* London: Taylor Graham, pp.115-129

Gould, J. D. (1987) Reading Is Slower from CRT Displays than Paper: Attempts to Isolate a Single-Variable Explanation. *Human Factors* 29(3), pp.269-299

Gould, J. D., Alfaro, L. , Finn, R., Haupt, B. and Minuto, A. (1987) Reading from CRT Displays Can Be as Fast as Reading from Paper. *Human Factors* 29(5), pp.497-517

Gould, J.D, Boies, S. J. and Lewis, C. (1991) Making Usable, Useful, Productivity-Enhancing Computer Applications. *Communications of the ACM* 34(1), January. pp.74-85

Graphic Communication Association (1992) *International Markup '92/Documentation Europe '92 Conference [Exhibitors Guide] 10-15 May, Amsterdam*

Haines, H.E. (1965) *Living with Books. The Art of Book Selection.* Second Edition. New York: Columbia University Press

Halasz, F. G. (1988) Reflections on Notecards: Seven Issues for the Next Generation of Hypermedia Systems. *Communications of the ACM* 31(7). pp.836-852

Hammond, N. and Allinson, L. (1987) The Travel Metaphor as a Design Principle and Training Aid for Navigating around Complex Systems, in, Diaper, D. and Winder, R. (eds). *People and Computers III.* Cambridge: Cambridge University Press, pp.75-90

Hansen, W. J. and Haas, C. (1988) Reading and Writing with Computers: a Framework for Explaining Differences in Performance, *Communications of the ACM* 31(9) September. pp.1080-1089

Hart, M. (1992) Private correspondence

Hildreth, C. R. (1985) Online Public Access Catalogs, in, Williams, M.E. (ed). *Annual Review of Information Science and Technology (ARIST)* Vol.20. Knowledge Industry Publications. pp.233-286

Hoffman, M.M., O'Gorman, L., Story, G.A., Arnold, J.Q. and Macdonald, N.H. (1993) The RightPages™ Service:An Image-Based Electronic Library. *Journal of the American Society for Information Science* 44(8) September. pp.446-452

Ichimura, S., Maeda, F. Kudo, M. and Matsushita, Y. (1993) OpenBook: a Hypermedia System Using the Book Metaphor. Transactions of the Information Processing Society of Japan 34(5). pp.1053-1063

IGBP (1991) *Global Change System for Analysis, Research and Training (START). Report of the Bellagio Meeting, December 1990.* Eddy, J.A., Malone, T.F., McCarthy, J.J. and Rosswall, T. (eds). Boulder: OIES, [IGBP Global Change Report No. 15]

IGBP (1991a) *Plant-Water Interactions in Large-Scale Hydrological Modelling. Report of a Workshop, Vadstena, Sweden, 5-8 June 1990.* Stockholm: IGBP, [IGBP Global Change Report No. 17]

IGBP (1992) *Report from the START Regional Meeting for Southeast Asia.* Stockholm: IGBP, [IGBP Global Change Report No. 22]

IIASA (1991) *Options*, [special issue on Global Change], December

Jackson, M.M. (1992) The Future of Books in an Electronic Era. *The International Information & Library Review* 24(4) December. pp.307-312

King, S.V. (1992) ELNET - the Electronic Library Database System, in, Raitt, D. (ed). *Online Information 92, 16th International Online Information Meeting Proceedings, London, 8-10 December 1992.* Oxford: Learned Information, pp.557-572

Kirstein, P.T. (1993) On-line Access to Journal Database Documents in Standardised Formats, in, *Document Interoperability Conference, Presentation Notes, 16-17 November 1993, London*

Lancaster, F.W., Drasgow, L.S. and Marks, E.B (1979) The Changing Face of the Library: a Look at Library and Librarians in the Year 2001. *Collection Management* 3. pp.55-77

Landauer, T., Egan, D., Remde, J., Lesk, M., Lochbaum, C. and Ketchum, D. (1993) Enhancing the Usability of Text through Computer Delivery and Formative Evaluation: the SuperBook Project, in, McKnight, C., Dillon, A. and Richardson, J. (eds) *Hypertext: a Psychological Perspective.* Chichester: Hellis Horwood

Lande, N. (1991) Toward the Electronic Book. *Publishers Weekly* Vol. 238(42). pp.28-30

Landoni, M., Catenazzi, N. and Gibb, F. (1993) Hyper-books and Visual-books in an Electronic Library. *The Electronic Library* 11(3) June. pp.175-186.

Large, A., Beheshti, J., Breuleux, A. and Renaud, A. (1992) Multimedia and the Acquisition of Information, in, Raitt, D. (ed), *Online Information '92, 16th International Online Information Meeting Proceedings, London, 8-10 December 1992.* Oxford: Learned Information, pp. 233-240

Lesk M. (1991) The CORE Electronic Chemistry Library, in, *Proceedings of the the Foutheenth Annual International ACM/SIGIR Conference on R&D in IR.* pp.92-112

Lie, H. and Bender, W. (1992) The Electronic Broadsheet: all the News that Fit the Display, in, Vanoirbeek, C. and Coray, G. (eds). *Electronic Publishing '92 International Conference on Electronic Publishing, Document Manipulation, and*

Typography, 7-10 April 1992, Lausanne, Switzerland. Cambridge: Cambridge University Press, pp.199-209

Liebing, E. (1993) Reading the Trend away from Paper: Here Are Some Tools to Help Manage your Electronic Documents. *LAN-Times* 10(20). pp.59-60

Lu, C. (1993) Publish It Electronically. *Byte* September. pp.94-109

Lucarella, D. (1990) A Model for Hypertext-Based Information Retrieval, in, Rizk A., Streitz N. and André, J. (eds) *Proceedings of the European Conference on Hypertext, Hypertext: Concepts, Systems and Applications, INRIA, France November 1990.* Cambridge: Cambridge University Press, pp.81-94

Lyon, E. (1990) Is There a Future for the Intermediary?, in, Armstrong, C.J. and Hartley, R.J. (eds). *Online Information Retrieval Today and Tomorow.* Oxford: Learned Information, pp.119-129

Macleod, I. A., Barnard, D.T., Hamilton, D. and Levison, M. (1991) SGML Documents and Non-linear Text Retrieval, in, *RIAO '91 Conference Proceedings: Intelligent Text and Image Handling, Barcelona Spain, 2-5 April 1991*, pp.226-244

Marchionini, G. (1989) Making the Transition from Print to Electronic Encyclopedia: Adaptation of Mental Models. *Int. J. Man-Machine Studies* 30(6) June. pp.591-618

Martin, D. (1989) *An Outline of Book Design.* London: Blueprint/The Publisher Association

McGarry, K. (1991) *Literacy, Communication and Libraries - A Study Guide.* London: Library Association Publishing

McKnight, C., Dillon, A. and Richardson, J. (1989) Problems in Hyperland? A Human Factor Perspective. *Hypermedia* 1(2). London: Taylor Graham, pp. 167-178

McKnight, C., Dillon, A. and Richardson, J. (1991) *Hypertext in Context.* P. Hammersley (ed). Cambridge: Cambridge University Press

Mittelbach, F. and Rowley, C. (1992) The Pursuit of Quality: How can Automated Typesetting Achieve the Highest Standards of Craft Typography?, in, Vanoirbeek, C. and Coray, G. (eds). *Electronic Publishing '92 International Conference on Electronic Publishing, Document Manipulation, and Typography, 7-10 April 1992, Lausanne, Switzerland.* Cambridge: Cambridge University Press, pp.261-273

Morris, A. (ed). (1992) *The Application of Expert Systems in Libraries and Information Centres.* London: Bowker-Saur, pp.34-67

Mynatt, B.T., Leventhal, L.M., Instone, K., Farhat , J. and Rohlman, D.S. (1992) Hypertext or Book: Which is Better for Answering Questions?, in, *CHI'92 Proceedings, May 3-7 1992.* ACM, pp.19-25

Newcomb, S.R., Kipp, N.A., Newcomb, V.T. (1991) The Hytime Hypermedia/time-based Document Structure Language. *Communications of the ACM* 34 (11). pp.67-83

Niblett, T. and van Hoff, A. (1989) Structured Hypertext Documents Via SGML. *Poster presentation at Hypertext II, University of York, June*

Nielsen, J. (1990) *Hyertext and Hypermedia.* Boston: Academic Press

Nielsen, J. (1990a) Through HYPERTEXT. *Communications of the ACM* 33(3). pp.297-310

Nielsen, J. and Molich, R. (1990) Heuristic Evaluation of User Interfaces, in, *CHI'90 Proceedings, April 1990*, ACM, pp.249-256

Parkinson, K.L. (1993) DynaText Publishing to Go to Mac; will Build SGML Files. *MacWEEK* 7(14)

Paulapuro, H. (1991) The Future of Paper in the Information Society. *The Electronic Library* 9(3) June. pp.135-143

Pejtersen, A.M. (1992) New Model for Multimedia Interfaces to Online Public Access Catalogues. *The Electronic Library* 10(6) December. pp.359-366

Quint, V. and Vatton, I. (1992) Combining Hypertext and Structured Documents in Grif, in, Lucarella, D., Nanard, J., Nanard, M. and Paolini, P. (eds). *Proceedings of the ACM Conference on Hypertext, Milano, Italy, 30 November - 4 December 1992.* New York: ACM. pp.23-32

Rada, R. and Murphy, C. (1992) Searching Versus Browsing, in, Hypertext. *Hypermedia* 4(1). London: Taylor Graham, pp.1-30

Raitt, D. I. (1985) Electronic Publishing - a View from the Library. *Electronic Publishing Review* 5(3). pp.199-210

Rawlins, G.J.E. (1993) Publishing Over the Next Decade. *Journal of the American Society for Information Science* 44(8). pp.474-485

Raymond, D.R. and Tompa, F.Wm. (1987) Hypertext and the Oxford English Dictionary, in, *Proceedings of Hypertext '87 Workshop, University of North Carolina, Chapel Hill, NC, 13-15 November 1987.*

Remde, J.R., Gomez, L.M. and Landauer, T.K. (1987) SuperBook: An automatic Tool for Information Exploration - Hypertext?, in *Proceedings of Hypertext '87 Workshop, University of North Carolina, Chapel Hill, NC, 13-15 November 1987.* pp.175-188

Reynolds, L.R. and Derose, S.J. (1992) Electronic Books. *Byte* 17(6) June. pp.263-268

Rosenthal, S. (1992) The Power to Read a Good Book. *MacWEEK* 6(6)

Rubinstein, R. (1988) *Digital Typography. An Introduction to Type and Composition for Computer System Design.* Reading: Addison-Wesley Publishing Company

Samuelson, P. and Glushko, R. J. (1991) Intellectual Property Rights For Digital Library and Hypertext Publishing Systems: An Analysis of Xanadu, in, *Proceedings of the Third ACM Conference on Hypertext, San Antonio, Texas, 15-18 December 1991*, New York: ACM Press, pp.39-49

Saunders, L. M. (ed). (1993) *The Virtual Library Vision and Reality.* Westport: Meckler

Shneiderman, B. (1992) *Designing the User Interface - Strategies for Effective Human-Computer Interaction.* Second Edition. Reading: Addison-Wesley Publishing Company

Singh, J. and Meadows, J. (1993) Electronic Serials for Library and Information Specialists on Internet, in, *Aslib Proceedings*, 45(9) September. pp. 234-243

Smith, J.M. (1987) *The Standard Generalized Markup Language (SGML): Guidelines for Authors.* Wetherby: British National Bibliography Research Fund [British National Bibliography Research Fund Report 27]

Smith, J.M. (1987a) *The Standard Generalized Markup Language (SGML): Guidelines for Editors and Publishers.* Wetherby: British National Bibliography Research Fund [British National Bibliography Research Fund Report 26]

Southall, R. (1992) Presentation Rules and Rules of Composition in the Formatting of Complex Text, in, Vanoirbeek, C. and Coray, G. (eds). *Electronic Publishing '92 International Conference on Electronic Publishing, Document Manipulation, and Typography, 7-10 April 1992, Lausanne, Switzerland.* Cambridge: Cambridge University Press, pp.275-290

Stotts, P. D. and Furuta, R. (1989) Petri-Net-Based Hypertext: Document Structure with Browsing Semantics. *ACM Transactions on Information Systems* 7(1) January. pp.3-29

Sullivan, P. (1989) Beyond a Narrow Conception of Usability Testing. *IEEE Transactions on Professional Communication* 32(4) December. pp.256-264

Tompa, F. WM. (1989) A Data Model for Flexible Hypertext Databases Systems. *ACM Transactions on Information Systems* 7(1) January. pp.85-100

Tuck, B. (1992) Approaches to Document Delivery. *Electronic Documents* 1(12). pp.12-16

Tuck, W.R. (1992a) Document Delivery and High-speed Networks, in, Raitt, D. (ed), *Online Information '92, 16th International Online Information Meeting Proceedings, London, 8-10 December 1992.* Oxford: Learned Information, pp.7-14

van Herwijnen, E. (1990) *Practical SGML.* Dordrecht: Kluver Academic Publishers

van Raaij, T. (1993) Portable, Device Independent Documents Hype or Reality, in, *Document Interoperability Conference, Presentation Notes, 16-17 November 1993, London*

Vickery, A., Brooks, H.M. and Robinson, B. (1987) A Reference and Referral System Using Expert Systems Techniques. *Journal of Documentation* 43(1). pp.1-23

Vogt, S. (1992) CD-ROM and Online - Competitors or Compatriots? An Online Host's Point of View, in, Raitt, D. (ed), *Online Information '92, 16th International Online Information Meeting Proceedings, London, 8-10 December 1992*. Oxford: Learned Information, pp.33-45

Walker, S. (1989) Online Catalogues. *Library and Information Briefings* June. pp.77-88

Wang, C. (1986) Electronic Publishing and its Impact on Books and Libraries: An Overview with Comment and Recommendation. *Electronic Publishing Review* 6(1). pp.43-55

Whiteside, J., Bennett, J. and Holtzblatt, K. (1988) Usability Engineering: Our Experience and Evolution, in, Helander, M. (ed). *Handbook of Human-Computer Interaction*. North Holland: Elsevier Science Publishers B. V., pp.791-817

Wood, L. (1992) Desktop Publishing Goes On line. *Datamation* 15 December. pp.45-47

Wright, H. (1992) SGML Frees Information. *Byte* 17(6) June. pp.279- 286

Wu, Z., Zhao, D. and Ramsden, A. (1993) From Automated Library to Electronic Library: Challenges for Information Retrieval, in, *Proceedings of British Computer Society Information Retrieval Group Annual Conference, Glasgow, 29-30 March 1993*. London: Taylor Graham [to be published]

Yankelovich, N., Meyrowitz, N. and Van Dam, A. (1985) Reading and Writing the Electronic Book. IEEE Computer 18(10) October. pp.15-30

Yankelovich, N. (1990) From Electronic Books to Electronic Libraries: Revisiting "Reading and Writing the Electronic Book", in, Delany, P. and Landow, G. (eds). *Hypermedia and Literary Studies*. Cambridge: The MIT Press, pp.133-141

Zadeh, L.A. and Polak, E. (1969) *System Theory*. New York: McGraw-Hill

Appendix A:

The Hyper-book Document Type Definition

```
<!doctype book
[
<!ENTITY % link "nlink | intlink | extlink | glink | biblink | dlink"     --links -- >
<!ENTITY % ientry "i1 | i2"                                              --index entry -- >

<!ELEMENT book - - (fm, mb, bm)>

<!ELEMENT fm - - (tipage, verso, (ab? & fr? & pr? & ak? & ded? & conlist &
    flist? & tlist?))>
<!ELEMENT tipage - - (ti & subti? & au* & ed* & pub? & cb* & edn? & pla* &
    da? & series? & infotip?)>
<!ELEMENT verso - - ( edn? & cip? & pubdet? & impdet? & pridet? & copdet? &
    ISSN? & ISBN* & infover?)>
<!ELEMENT (ab | fr | pr | ak | ded) - - (par+) -(%ientry; )>
<!ELEMENT (au | pub | ed | cb) - - (name, add?)>
<!ELEMENT series - - (serna?, sernu?)>
<!ELEMENT (conlist | flist | tlist) - o EMPTY  -- content, figure, table list -- >
<!ELEMENT (ti | subti | edn | cip | pla | da | pubdet | impdet | pridet | copdet | ISBN |
    ISSN | infotip | infover) - - (#PCDATA)>
<!ELEMENT (name | p | add | serna | sernu) - - (#PCDATA)>

<!ELEMENT mb - - (chap+)                                    -- main body -- >
<!ELEMENT chap - - (h1t, par*, sec2*) +(fnote | graph )>
<!ATTLIST chap id ID #REQUIRED>
<!ELEMENT sec2 - - (h2t, par*, sec3*) >
<!ATTLIST sec2 id ID #REQUIRED>
<!ELEMENT sec3 - - (h3t, par*, sec4*) >
<!ATTLIST sec3 id ID #REQUIRED>
<!ELEMENT sec4 - - (h4t, par*) >
<!ATTLIST sec4 id ID #REQUIRED>
<!ELEMENT (h1t | h2t | h3t | h4t) - - (#PCDATA)>
<!ELEMENT par - - (#PCDATA | %ientry; | %link;)+ >

<!ELEMENT fnote - - (p+)                    -- footnotes -- >
<!ATTLIST fnote id ID #REQUIRED>
```

```
<!ELEMENT graph - - (gcap?)            -- graphics -- >
<!ELEMENT gcap - - (par+) -(%ientry; )>
<!ATTLIST graph      id ID #REQUIRED
                     filename CDATA #REQUIRED
                     type (TablelFigure) "Figure"
                     width NUMBERS #REQUIRED
                     height NUMBERS #REQUIRED>

<!ELEMENT (i1 | i2) - - (#PCDATA)>
<!ATTLIST i1    id ID #IMPLIED
                seeid IDREF #IMPLIED>
<!ATTLIST i2    id ID #IMPLIED
                parid IDREF #REQUIRED
                seeid IDREF #IMPLIED>

<!ELEMENT (nlink | biblink) - o  EMPTY>
<!ELEMENT (glink | intlink | extlink | dlink) - -  (#PCDATA)>
<!ATTLIST (nlink | biblink | glink | intlink | dlink) rid IDREF #REQUIRED>
<!ATTLIST extlink filename CDATA #REQUIRED
                  target CDATA #IMPLIED>

<!ELEMENT bm - -  ((ref? & index? & glos? & rdoc? & acr? & bio?), ap?)
                                        -- back matter -- >
<!ELEMENT ap - -  (chap*)>
<!ELEMENT ref - -  (bi+)>
<!ELEMENT rdoc - -  (rdoci+)>
<!ELEMENT index - o EMPTY>
<!ELEMENT glos - - (gloterm, glosdef)+>
<!ELEMENT acr - - (acterm, acdef)+>
<!ELEMENT (acterm | acdef | gloterm | bi | rdoci | bio) - -  (#PCDATA)>
<!ATTLIST (gloterm | acterm) id ID #IMPLIED>
<!ELEMENT glosdef - -  (p+) +(link)>
<!ATTLIST (bi | rdoci) id ID #REQUIRED>
]>
```

Appendix B:
The Evaluation

PERSONAL DETAILS

Name:

Address/Contact:

Qualification:

USER EXPERIENCE

- How often do you use a computer?

- never ☐
- occasionally ☐
- roughly once per week ☐
- roughly once per day ☐
- spend most of my working day on computers ☐

- How well do you know Apple Macintosh?

- do not know ☐
- just word processors ☐
- competent with one programme ☐
- competent with a few programmes ☐
- know it well ☐

- How would you assess your knowledge of Global Change?

- none ☐
- limited ☐
- reasonable ☐
- good ☐
- very good ☐

- How familiar are you with scientific English?

very familiar not familiar at all

1 2 3 4 5

Tasks

1. BROWSING

Indicate if and where the following topics are discussed in the collection (specify the report and the section numbers):

1.1 - air chemistry

1.2 - training of researchers

1.3 - upscaling

1.4 - ecological education

2. SEARCHING

Use report 15 for answering the following questions:
(specify section and page numbers if the answer is too long)

2.1 - What was the purpose of the meeting in Bellagio?

2.2 - Why has a geographical subdivision been required?

2.3 - What is the meaning and the main role of a RRC?

2.4 - How is a RRC organized?

2.5 - Have the national limits been taken into account while choosing the regions?

2.6 - Which regions have been proposed by the START initiative?

2.7 - Which region(s) does Europe belong to?

2.8 - What's the meaning and the role of SC-IGBP?

2.9 - In which report are the Core Projects described?

2.10 - Did Turner participate at the workshop in Bellagio?

3. ANALYSIS

3. Let us suppose that you have to use these documents as a reference material for a paper. The subject of the paper is the climatic conditions of the different regions proposed by the START Initiative. Analyse the topic, selecting and highlighting the most relevant parts for a future use.

Questionnaire[1]

USABILITY

- How easy was the system to learn?

	very easy				very difficult
	1	2	3	4	5

- Rate the ease of use and understanding of the following tools:

- add / delete bookmark	very easy				very difficult
	1	2	3	4	5
- add / delete note	very easy				very difficult
	1	2	3	4	5
- annotations	very easy				very difficult
	1	2	3	4	5
- copy text	very easy				very difficult
	1	2	3	4	5
- return	very easy				very difficult
	1	2	3	4	5
- recent	very easy				very difficult
	1	2	3	4	5
- use of bookmarks	very easy				very difficult
	1	2	3	4	5
- links from Table of Contents/	very easy				very difficult
List of Tables / List of Figures	1	2	3	4	5

[1] the purpose of the questionnaire is to assess user's opinion about the system functionality; the intention is not to consider system efficiency.

- links in the text	very easy				very difficult
	1	2	3	4	5
- zoom property of pictures	very easy				very difficult
	1	2	3	4	5
- go to the next/previous page	very easy				very difficult
	1	2	3	4	5
- flip pages	very easy				very difficult
	1	2	3	4	5
- go to page	very easy				very difficult
	1	2	3	4	5
- thickness	very easy				very difficult
	1	2	3	4	5
- go to Table of Contents	very easy				very difficult
	1	2	3	4	5
- go to List of Figures/Tables	very easy				very difficult
	1	2	3	4	5
- go to Index	very easy				very difficult
	1	2	3	4	5
- go to front/back cover	very easy				very difficult
	1	2	3	4	5
- print	very easy				very difficult
	1	2	3	4	5
- open a word processor	very easy				very difficult
	1	2	3	4	5

- searching very easy very difficult
1 2 3 4 5

- Did you have any recurring errors?
If so, where did they occur?

- How easy was correcting your mistakes?
very easy very difficult
1 2 3 4 5

- Did the way information was presented ever mislead you ?
If so, where?

- How much did you enjoy using the system?
enjoyed it a lot found it very frustrating
1 2 3 4 5

- How much did you enjoy using a book presented in this way?
enjoyed it a lot found it very frustrating
1 2 3 4 5

UTILITY

- Were all the functions you would like to use adequately supported ? If not, which functions need to be added/extended?

- Rate the utility of the following tools:

- notes	very useful				not at all useful
	1	2	3	4	5
- annotations	very useful				not at all useful
	1	2	3	4	5
- bookmarks	very useful				not at all useful
	1	2	3	4	5
- return	very useful				not at all useful
	1	2	3	4	5
- recent	very useful				not at all useful
	1	2	3	4	5
- links from Table of Contents / List of Tables /List of Figures	very useful				not at all useful
	1	2	3	4	5
- links in the text	very useful				not at all useful
	1	2	3	4	5
- zoom property of pictures	very useful				not at all useful
	1	2	3	4	5
- go to the next/previous page	very useful				not at all useful
	1	2	3	4	5

- flip pages	very useful				not at all useful
	1	2	3	4	5
- go to page	very useful				not at all useful
	1	2	3	4	5
- go to Table of Contents	very useful				not at all useful
	1	2	3	4	5
- go to List of Figures/Tables	very useful				not at all useful
	1	2	3	4	5
- go to Index	very useful				not at all useful
	1	2	3	4	5
- go to front/back cover	very useful				not at all useful
	1	2	3	4	5
- thickness	very useful				not at all useful
	1	2	3	4	5
- print	very useful				not at all useful
	1	2	3	4	5
- open a word processor	very useful				not at all useful
	1	2	3	4	5
- searching	very useful				not at all useful
	1	2	3	4	5

- Do you think that this kind of electronic text presentation is useful for accomplishing your tasks?

If yes, explain why.

If not, why not?

- Would you use an electronic book?

If the answer is yes, explain in which situations. If not, explain why not.

Appendix C:
The Evaluation Results

Quality

Browsing Task

Group 1

TASK ID	User 1	User 2	User 3	User 4	User 5	User 6	Group Quality
1.1	2	2	2	0	0	2	1.33
1.2	2	2	1	1	1	2	1.50
1.3	1	0	0	2	1	0	0.67
1.4	2	2	0	2	2	2	1.67

Group 2

TASK ID	User 1	User 2	User 3	User 4	User 5	User 6	Group Quality
1.1	2	2	2	2	2	2	2.00
1.2	2	1	1	0	2	2	1.33
1.3	2	2	2	0	0	0	1.00
1.4	2	0	0	2	0	2	1.00

Searching Task

Group 1

TASK ID	User 1	User 2	User 3	User 4	User 5	User 6	Group Quality
2.1	1	2	1	2	1	2	1.50
2.2	2	0	2	2	2	2	1.67
2.3	1	2	2	2	2	2	1.83
2.4	1	2	2	2	2	2	1.83
2.5	0	2	2	0	0	2	1.00
2.6	2	2	2	2	2	2	2.00
2.7	2	1	2	2	2	2	1.83
2.8	2	2	1	1	2	2	1.67
2.9	0	0	2	2	2	2	1.33
2.10	2	2	2	2	2	2	2.00

Group 2

TASK ID	User 1	User 2	User 3	User 4	User 5	User 6	Group Quality
2.1	1	1	1	1	1	2	1.17
2.2	1	2	0	1	0	1	0.83
2.3	2	2	2	2	2	2	2.00
2.4	0	2	2	2	2	2	1.67
2.5	2	0	0	2	2	2	1.33
2.6	2	2	2	2	2	2	2.00
2.7	1	1	2	1	1	1	1.17
2.8	0	2	1	2	2	2	1.50
2.9	2	2	2	0	2	2	1.67
2.10	2	0	2	2	0	2	1.33

Time

Browsing Task

Group 1

TASK ID	User 1	User 2	User 3	User 4	User 5	User 6	Group Time
1.1	2	2	3	3	4	2	2.67
1.2	3	2	1	0	3	4	2.17
1.3	3	2	2	2	3	1	2.17
1.4	3	3	4	4	3	3	3.33

Group 2

TASK ID	User 1	User 2	User 3	User 4	User 5	User 6	Group Time
1.1	2	2	3	1	4	3	2.50
1.2	2	2	3	1	2	4	2.33
1.3	2	1	3	1	2	1	1.67
1.4	3	3	4	1	3	2	2.67

Searching Task

Group 1

TASK ID	User 1	User 2	User 3	User 4	User 5	User 6	Group Time
2.1	2	2	3	2	2	2	2.17
2.2	1	3	3	3	1	3	2.33
2.3	1	2	2	1	1	2	1.50
2.4	3	3	1	2	2	1	2.00
2.5	2	3	1	1	4	1	2.00
2.6	1	1	1	1	0	1	0.83
2.7	3	1	1	1	1	2	1.50
2.8	2	3	4	4	3	4	3.33
2.9	4	4	2	0	3	1	2.33
2.10	1	3	2	1	1	1	1.50

Group 2

TASK ID	User 1	User 2	User 3	User 4	User 5	User 6	Group Time
2.1	2	1	4	4	2	2	2.50
2.2	1	1	4	3	4	4	2.83
2.3	2	1	3	2	1	4	2.17
2.4	1	2	3	3	1	2	2.00
2.5	2	1	3	3	1	2	2.00
2.6	1	3	1	2	1	1	1.50
2.7	1	1	2	1	1	0	1.00
2.8	4	3	2	1	3	2	2.50
2.9	4	1	2	4	2	2	2.50
2.10	1	2	4	1	0	1	1.50

Frequency of tool usage

Browsing Task

Tool group ID	Group 1		Group 2	
	Absolute number	%	Absolute number	%
HL	20	6	16	7
TL	2	1	7	3
LA	76	21	47	21
DIA	35	10	72	32
DEA	41	12	39	17
H	3	1	1	0
ES	0	0	0	0
S	179	50	45	20
P	0	0	0	0

Searching Task

Tool group ID	Group 1		Group 2	
	Absolute number	%	Absolute number	%
HL	37	8	38	11
TL	40	9	31	9
LA	131	29	123	35
DIA	62	14	64	18
DEA	0	0	0	0
H	1	0	7	2
ES	0	0	0	0
S	188	41	80	23
P	0	0	7	2

Analysis Task

Tool group ID	Group 1		Group 2	
	Absolute number	%	Absolute number	%
HL	11	10	5	4
TL	2	2	1	1
LA	29	28	57	41
DIA	13	12	12	9
DEA	0	0	0	0
H	1	1	3	2
ES	23	22	13	9
S	0	0	3	2
P	26	25	46	33

Final Questionnaire

Usability

How easy was the system to learn?.

Group	User 1	User 2	User 3	User 4	User 5	User 6	Group Average
Group1	2	1	3	1	1	1	1.5
Group 2	1	1	1	1	2	1	1.17

Rate the ease of use and understanding of the following tools

Group 1

Tool	User 1	User 2	User 3	User 4	User 5	User 6	Group Average
add/delete bookmark	2	1	2	1	1	1	1.33
add/delete note	2	1	2	1	1	1	1.33
annotation	3	1	2	1	3	1	1.83
copy text	2	1	3	1	3	1	1.83
return	3	1	2	1	3	1	1.83
recent	2	1	2	1	2	1	1.50
use of bookmarks	2	1	3	1	1	1	1.50
links from Contents, List of Tab./Fig.	1	1	2	1	1	1	1.17
links in the text	2	1	3	1	1	1	1.50
zoom of pictures	2	1	2	1	1	1	1.33
go next/previous page	2	1	2	1	1	1	1.33
flip pages	2	1	2	1	3	1	1.67
go to page	1	2	3	1	1	1	1.50
thickness	2	1	2	2	2	1	1.67
go to Table of Contents	1	1	3	1	2	1	1.50
go to List of Figures/Tables	1	1	3	1	2	1	1.50
go to Index	1	1	3	1	2	1	1.50
go to back cover/front cover	1	1	2	1	1	1	1.17
print	4	1	3	1	1	1	1.83
open a word processor	3	1	3	1	1	1	1.67
searching	2	1	4	1	3	1	2.00

Group 2

Tool	User 1	User 2	User 3	User 4	User 5	User 6	Group Average
add/delete bookmark	1	1	1	1	1	1	1.00
add/delete note	1	1	2	1	1	1	1.17
annotation	1	1	2	1	1	2	1.33
copy text	1	1	1	1	2	1	1.17
return	1	1	2	1	1	1	1.17
recent	1	1	1	2	1	1	1.17
use of bookmarks	1	1	1	1	1	1	1.00
links from Contents, List of Tab./Fig.	1	2	1	1	1	1	1.17
links in the text	1	2	2	1	1	1	1.33
zoom of pictures	1	1	1	1	1	1	1.00
go next/previous page	1	1	1	1	1	1	1.00
flip pages	1	1	1	1	1	1	1.00
go to page	1	1	1	1	1	1	1.00
thickness	1	1	1	1	1	1	1.00
go to Table of Contents	1	1	1	1	1	1	1.00
go to List of Figures/Tables	1	1	1	1	1	1	1.00
go to Index	1	1	1	1	1	1	1.00
go to back cover/front cover	1	1	1	1	1	1	1.00
print	1	1	2	1	1	1	1.17
open a word processor	1	2	2	1	1	1	1.33
searching	1	1	2	1	1	1	1.17

How much did you enjoy using the system?

Group	User 1	User 2	User 3	User 4	User 5	User 6	Group Average
Group1	2	1	3	2	2	1	1.83
Group 2	1	1	1	1	1	2	1.17

How much did you enjoy using the book presented in this way?

Group	User 1	User 2	User 3	User 4	User 5	User 6	Group Average
Group1	2	1	3	2	1	1	1.67
Group 2	1	1	1	1	3	2	1.50

Utility

Group 1

Tool	User 1	User 2	User 3	User 4	User 5	User 6	Group Average
go next/previous page	1	1	2	1	1	1	1.17
go to page	1	1	2	1	1	1	1.17
go to Table of Contents	1	1	2	1	1	1	1.17
go to Index	1	1	2	1	1	1	1.17
print	2	1	1	1	1	1	1.17
links from Contents, List of Fig./Tab.	1	1	2	1	2	1	1.33
go to List of Figures/Tables	1	1	2	1	2	1	1.33
open a word processor	2	2	2	1	3	1	1.83
searching	1	1	2	1	1	1	1.17
links in the text	1	1	2	1	2	1	1.33
zoom of pictures	2	1	2	1	2	1	1.50
return	1	2	3	1	4	4	2.50
bookmarks	1	1	3	1	1	4	1.83
notes	1	1	2	1	1	1	1.17
flip pages	2	3	4	1	2	1	2.17
recent	1	2	3	1	4	4	2.50
go to back/front cover	2	1	3	3	4	1	2.33
annotations	1	1	3	2.5	1	1	1.58
thickness	2	2	2	2	2	4	2.33

Group 2

Tool	User 1	User 2	User 3	User 4	User 5	User 6	Group Average
go next/previous page	1	1	1	1	1	1	1.00
go to page	1	1	1	1	1	1	1.00
go to Table of Contents	1	1	1	1	1	1	1.00
go to Index	1	1	1	1	1	1	1.00
print	1	1	1	1	1	1	1.00
links from Contents, List of Fig./Tab.	1	1	1	1	1	1	1.00
go to List of Figures/Tables	1	2	1	1	1	1	1.17
open a word processor	1	2	1	1	1	1	1.17
searching	1	1	1	3	1	1	1.33

links in the text	1	1	1	3	1	1	1.33
zoom of pictures	1	1	3	1	1	1	1.33
return	1	1	1	1	4	1	1.50
bookmarks	1	3	1	2	3	2	2.00
notes	2	1	2	2	4	2	2.17
flip pages	2	1	4	1	3	4	2.50
recent	2	3	1	5	3	1	2.50
go to back/front cover	3	3	3	1	2	4	2.67
annotations	2	3	1	4	4	3	2.83
thickness	3	2	3	3	2	4	2.83