

# **Book4All: A tool to make an e-book more accessible to students with vision/visual-impairments**

Antonello Calabrò, Elia Contini and Barbara Leporini

ISTI - CNR

Via G. Moruzzi, 1 – 56124 Pisa, Italy

{antonello.calabro, elia.contini, barbara.leporini}@isti.cnr.it.

**Abstract.** Empowering people who are blind or otherwise visually impaired includes ensuring that products and electronic materials incorporate a broad range of accessibility features and work well with screen readers and other assistive technology devices. This is particularly important for students with vision impairments. Unfortunately, authors and publishers often do not include specific criteria when preparing the contents. Consequently, e-books can be inadequate for blind and low vision users, especially for students. In this paper we describe a semi-automatic tool developed to support operators who adapt e-documents for visually impaired students. The proposed tool can be used to convert a PDF e-book into a more suitable accessible and usable format readable on desktop computer or on mobile devices.

**Keywords:** e-book, e-document, accessibility, accessible publishing, semi-automatic support, DAISY

## **1 Introduction**

Accessing information as well as reading educational materials has always been a challenge for people unable to read printed material. Electronic materials are gaining more and more importance in daily life for everybody [9], especially for people with vision impairments. As technology advances and more books move from hard-copy print to electronic formats, people with print disabilities will have a great opportunity to enjoy access to books on an equal basis with those who can read print. This is particularly important for blind students (e.g. [4] and [8]).

To assure this is the case, associated mechanisms for guaranteeing accessibility as well as usability for all are required. Furthermore, in several countries, specific accessibility laws have been approved, which require publishers to provide electronically accessible versions of school textbooks. Nevertheless, students with disabilities in mainstream classrooms do not always have access to the same learning tools as their classmates. For example, students with visual impairments rely on alternative-format books (such as large or Braille print or audio versions).

Unfortunately, the time-consuming process of turning books or other materials into Braille, audiotape, or large-print editions means that visually impaired students often start the school year without their textbooks. Thus, while the other students receive

complete printed books, vision-impaired students receive their study material chapter by chapter with the risk that it does not arrive in time for them to keep up with the learning schedule. Moreover, visual learning tools are often totally unsuitable for blind students. This means that a special adapting process to make them accessible to students with vision impairments can be labour intensive. As a consequence, students must rely on teachers or classmates, who have to try to describe visual content as best they can. This situation can limit the educational opportunities of blind students.

We use the term e-document for contents (text, figures etc.) that is stored in electronic format. Thus, an e-document can be a simple text, a more complex document, with title, paragraphs and subparagraphs, or an electronic book organized into chapters and sections. In other words, all these considerations and the solutions we propose are relevant to a general e-document as well as to more highly structured e-books. Starting with these considerations, in the following we use the terms e-document and e-book synonymously.

The format of a large of e-documents available to visually impaired people can be fairly simple and easy to modify to provide accessibility, as is often the case for story books. However, scientific e-documents for example may contain complicated explanations, graphs, tables and formulae. In this case it is more difficult to create accessible and usable texts, since a complex network of alternative explanations and other complementary descriptions need to be included in the electronic format.

Overall, our work is aimed at addressing the accessibility and usability issues in electronic publishing. Specifically, it aims to develop technical and methodological solutions to the main barriers to creating cultural/scientific e-documents that can be accessed and used able by disabled users. In addition this work is intended to provide recommendations for the creation of accessible e-documents as well as support for converting inaccessible e-documents to an accessible format.

In this paper, first we introduce the main issues related to e-book user experience by considering both end-users – i.e. blind students – and adapting-center operators. Next, we describe our approach by showing how it has been developed. For this purpose, we describe the solution applied to two e-documents available on line.

## **2 Related work**

In order to overcome limitations for those people who cannot meet their information needs from standard printed material, electronic documents can represent a very appropriate solution. In this direction several attempts have been made to develop specialised formats, especially for people with a visual impairment. One of these is a by-product of the Daisy consortium (Digital Accessible Information SYstem), which is working on producing a standard for talking books characterized by navigational information to better move through the document (<http://www.DAISY.org>). With this system, hybrid books can be made available in the DAISY format, ranging from audio-only, to full text and audio, to text-only [3]. In November 2008, a new version of guidelines for DAISY standard 3.0 has been published [7]. These Guidelines provide information on the correct usage and application of DAISY XML (the DTBook XML element set) in the creation of DAISY publications. Another major change in the DAISY 3 Structure Guidelines is the inclusion of information about NIMAS (National Instructional Materials Accessibility Standard) mark-up [16]. NIMAS is the required electronic format in the U.S. for textbook publishers which produce K-12 textbooks.

Other studies on this topic include the work of Sun et al. [17], who designed e-Book browsers that aid users in perceiving and understanding the important conceptual structures of a book, and hence improve their comprehension of the book content. A number of authors have proposed ad-hoc solutions for visually-impaired people. Adjouadi et al. [1] introduced a new automatic book reader for blind people. Their objective is the design of a fully integrated system that is relatively fast, but inexpensive and effective with a high reading accuracy. Scientific contents present additional difficulties in reading through a screen reader. Kanahori et al. [14] proposed an integrated system for scientific documents including mathematical formulae with speech output interface. “ChattyInfty” is a system composed of a recognizer, editor and speech interface for scientific documents, including PDF documents. The system tries to recognise the content from the PDF as it was a page image and provides the interpreted content through a simple and customizable interface. Using this system, visually impaired people can read printed scientific documents with speech output.

In our study we prefer to propose solutions to design, develop or adapt accessible and usable e-books, which can be read with traditional tools (e.g. adobe reader, Web browser) in order to obtain more general solutions, with the greatest benefits for all. In this sense, we consider solutions aimed at converting and adapting e-documents in order to make them more usable for blind people.

The large amount of information available today can be potentially accessed in real time through intranet or internet. This has increased the need for syntactic and semantic characterization of documents and for tools that allow their effective access and exploitation on the Net as well as their annotation to adapt and personalize them on the base of users’ characteristics and diversities. In [2] the authors propose a system aimed at improving e-document editing, annotating, and at exploiting the description of the logical structure of the document itself to squeeze the information

about the document content. In [5] an approach for automatically generating an e-textbook on the Web for a user specified topic hierarchy is described. That approach consists of generating an e-book on the Web starting with a topic indicated by the user. Users can then browse through the descriptive pages like a book. Hence, e-books are increasingly providing new ways to obtain and study information. In [11] the problem related to tagged PDF is faced by using an XML-based structure. The tool implemented is an Acrobat Plugin, which means that it can be used in conjunction with Acrobat products. Conversely our methodology is aimed at developing a semi-automatic support to adapt a single format to various ones by taking into account specific user requirements.

### **3 The e-book User Experience (UX)**

In order to understand the main issues related to the interaction with an e-book, let us first consider the user experience when handling an e-book itself. Two different perspectives can be identified:

- The end-user viewpoint;
- The adapting-center viewpoint.

As end-users we consider blind people and in particular students who need to use books for educational purposes. The adapting-center viewpoint is provided by the operator who has to manage the content to make it accessible.

#### **3.1 End-users' Viewpoint**

Even though school books are available in a digital format such as TXT, XHTML or PDF, this fact alone is not enough to guarantee accessibility and usability features to support the studying process. Students should be able to read books not exclusively in a sequential way, as with story books: they should be able to move quickly through the content as well as to take notes and to mark interesting parts. In short, blind students should have the same opportunities to interact with an e-book as a sighted student has with the paper format.

Currently available e-books do not enable students to perform this kind of task, because the contents are not well structured, images do not have textual descriptions, data tables do not have text explanation, and it is difficult to mark parts or take notes. Consequently, these issues increase the learning effort and the cognitive load by the students as well. In fact, when the students need to revise specific parts, they have to read the contents in sequential way or try to find the required text by searching for a certain string or sentence. This is not practical when studying, because the students can lose focus while searching for the text. More details of blind people's experiences with e-documents can be found in [6] and [13]. In order to enhance the learning process, the contents should not only be accessible, but also – and equally importantly – usable [15]. This means that appropriate mechanisms and adequate descriptions need to be available in the e-book.

### **3.2 Adapting-center Viewpoint**

Nowadays, a serious problem is the slow and expensive production of e-books that are both accessible and for all, including people who have to interact with assistive technologies (e.g. a screen reader). To obtain a concrete accessible and usable e-book, the contents and the structure have to conform to appropriate procedure and rules. The e-book content is adapted by a person - called an operator -. Carrying out those actions manually is time-consuming and requires a lot of actions by the operators. The main reasons for this limitation can be summarized as follows:

- lack of clear guidelines that help authors/publishers to build a really accessible and usable e-book;
- lack of tools able to support the entire conversion process;
- lack of a standard intermediate format that makes it possible abstract data book content and that is not limited to final distribution formats such as PDF, XHTML, HTML5, DAISY 3.0 and Braille;
- Lack of consideration by publishers of blind and visually impaired student needs.

The work carried out is mainly focused on producing accessible versions of the textbooks received by the publishers possibly in electronic format. Simplifying the work of the adaptors in translating contents into a more adequate version that can be easily used via a screen reader would be an important contribution to rectify this limitation.

## **4 Our approach**

In order to solve problems related to e-book UX we propose a very practical approach with the following components:

- A set of guidelines that allows users and adapting-centers to obtain an excellent e-book UX;
- Semi-automatic tools to speed up the process of e-book conversion and adaptation (ad-hoc PDF viewer and Book4All);
- A very powerful and flexible Intermediate Book Format (IBF) to manage contents of e-books and provides a solid base to generate accessible and usable final formats.

Currently we provide a simple way to generate XHTML 1.0 Strict or DAISY 3.0 e-book.

The solution proposed in this work is an evolution of our previous case study [6]. Based on our previous experience, we are proposing a new and updated methodology. In particular, we selected two documents available on line ([9] [18]) in order to show how our approach can be applied to practical cases.

#### **4.1 Guidelines**

The generation of accessible and usable e-books is closely linked to a well structured content: each element within a book must be marked with the right tag and each complex element such as images and tables must be described using an alternative description.

When generating the e-book UI to be provided to the blind students, we consider five main guidelines:

1. Titles and subtitles (i.e. sections and subsections) must be marked with the appropriate tags. If titles are tagged it is possible to automatically generate the content index and to provide a basic and non-sequential navigation structure;
2. Paragraph and elements within paragraph (e.g. links, abbreviations and so on) must be marked. If paragraphs are tagged then it is possible to easily search the content of e-book, take notes and create bookmarks;
3. images must be tagged and each image must have an appropriate textual description that clearly explains the information provided;
4. tables must be tagged and each table must have an appropriate textual description that clearly explains the structure of table (e.g. Table has two columns) and the data contained in it (e.g. The first column contains users' names and the second one email);
5. Unordered and ordered lists must be tagged to improve navigation between items and to better identify each list (e.g. by using screen reader special command, the user is able to skip from one list to another).

#### **4.2 Semi-automatic Tools**

Our approach also includes two tools aimed at supporting adapting-centers to accelerate the conversion and adaptation process. Based on the consideration that book editors mainly provide e-book in PDF format, we consider this format as the source version. Consequently, our solution is based on a process that takes a PDF document as input file and provides an accessible and usable final electronic format as output, following various semi-automatic manipulations (the intervention of an operator is necessary).

##### **4.2.1 PDF Viewer**

In our case study, the first tool we created was an ad-hoc PDF Viewer that allows operators to gather useful data on the PDF source. This is necessary to obtain as much correct information as possible from the source version in order to facilitate the operators in manipulating the content to be adapted. As a consequence, an important assumption we made is about the presence of the operator in the conversion process. This means that (as described below) the more input information provided to the manipulation tool before starting the conversion process, the better the obtained results are. From this prospective, the user (i.e. the operator) could be asked to input some information on the PDF source document. The main goal of our proposed PDF Viewer is to provide styles and formats features as well as select and copy parts from

the entire content. For example, the PDF Viewer obtains information on the fonts such as name and size of the selected content.

In addition, with this supporting tool it is possible to manage PDF with multiple column layout. The selection of text in a multiple column layout PDF is a problem recently considered also by Apple Inc, which used artificial intelligence and implemented it in the new version of the Preview tool<sup>1</sup>. Our ad-hoc PDF Viewer provides a simpler and more detailed solution than the one proposed by Apple. And in fact, our tool also provides information about the selected text.

Figure 1 shows an example of multi-column text extraction with Ad-Hoc PDFViewer; in the popup window we have information about font face and size.

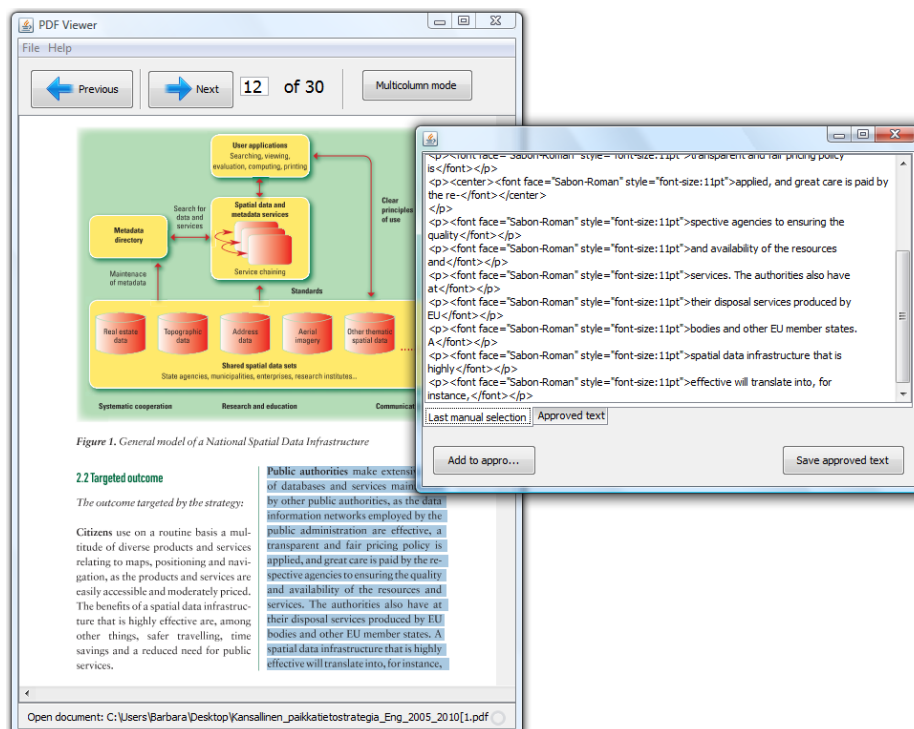


Fig. 1. Ad-hoc PDF Viewer. Text extraction from a multi-column layout document.

#### 4.2.2 BOOK4All

The information gathered about fonts is passed to the second tool “Book4All” which uses such information to extract text and images from the PDF source. Extracting information from a PDF document is a critical issue to be implemented. Similarly to our approach, in [12] a possible method developed to obtain text and images from a PDF file by analyzing the layout structure is proposed. Book4All reconstructs the

<sup>1</sup> Apple Preview Tool, <http://www.apple.com/macosx/refinements/enhancements-refinements.html#preview>

semantic structure of the e-book by analyzing the extracted data and saves the result in the Intermediate Book Format (IBF) (see 4.3). Each element is tagged according to the guidelines described above (see 4.1). After this step, it is possible to edit contents, add images and table descriptions and finally to export the result in one of the final supported formats (i.e. XHTML 1.0 or DAISY 3.0).

### 4.3 Intermediate Book Format (IBF)

The key issue of our solution is the Intermediate Book Format (IBF) that enables the user to tag elements of the e-book in an easy way. Overall, this XML-based format is designed to abstract the content of e-book and make such content independent from the final e-book format provided to the final users, the students.

IBF is a XML based language with an easy syntax that provides a tag for each element usually contained within a school textbook. At the current time, we can provide support only for simple text books: the work to obtain support for scientific books, such as mathematics, chemistry or physics, is in progress.

The following example shows a fragment of IBF code:

```
<page number="1">
<headingLevel1>Chapter title</headingLevel1>
<paragraph>Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Quisque non felis sed augue aliquam
eleifend nec ac lectus. Donec tempus, quam vitae
euismod ullamcorper, nisi nisl malesuada nulla,
<emphatyzed>a adipiscing nunc dolor et
nunc</emphatyzed>. Nullam eleifend, arcu sed
ullamcorper mollis, justo velit dapibus leo, sed
fringilla nisi arcu vel nulla. <strong>Sed euismod
dapibus tortor ac bibendum</strong>.
</paragraph>
<headingLevel2>Paragraph Title</headingLevel2>
<list type="ordered">
<listItem>Item 1</listItem>
<listItem>Item 2</listItem>
<listItem>
<list type="unordered">
<listItem>
<paragraph>Lorem ipsum dolor sit amet,
consectetur adipiscing elit. Phasellus a vestibulum
justo. <image source="image.png" width="88" height="33">
```



```
description="image description" />
    </paragraph>
  </listItem>
</list>
</listItem>
</list>
</page>
```

#### **4.4 A typical workflow**

Based on our procedure, a typical workflow would be the following:

1. The Editor provides a PDF school textbook ;
2. The operator uses ad-hoc PDF Viewer to gather information about fonts and make the content linear in case of multiple column layout (see 4.2);
3. The operator uses Book4All to extract text and images and automatically mark all text elements (title, paragraph and so on) with the right tag, by using the fonts information gathered by ad-hoc PDF Viewer
4. After the tagging step, the operator adds descriptions to images, corrects possible extraction and tagging mistakes (using particular reviews lists), and reconstructs the table manually;
5. At this point, the obtained IBF file is ready to be exported in one of the final formats;
6. The final format can be provided to students.

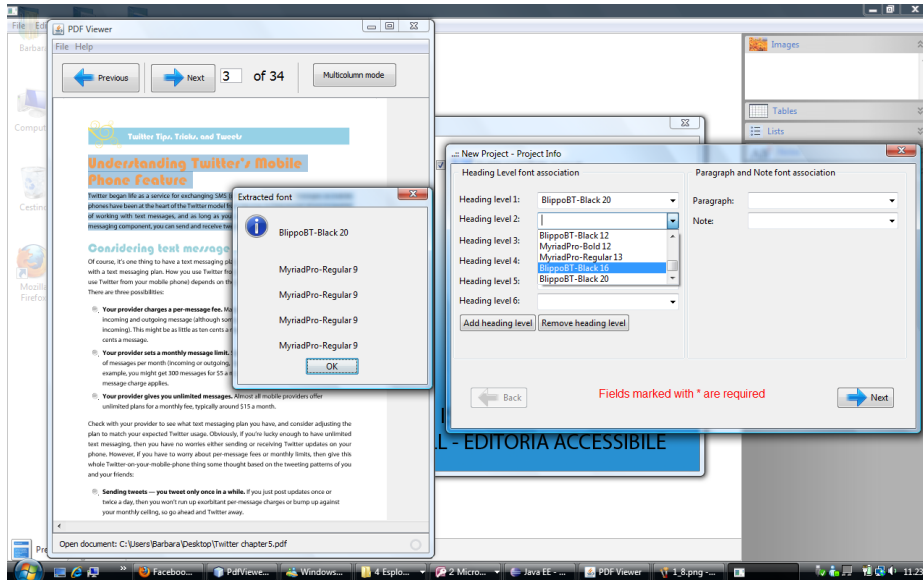


Fig. 2. Association of a font with structure elements with ad-hoc PDF Viewer aid.

## 5 Architecture

As mentioned above our solution is composed of two main tools: an ad-hoc PDF Viewer and Book4All based on the IBF API 1.0 (Stone Eater).

The IBF API 1.0 is the core of our architecture and it is completely GUI independent. Fig. 3. shows the main infrastructure based on our solution. This means that developers can create both Desktop GUIs and web applications.

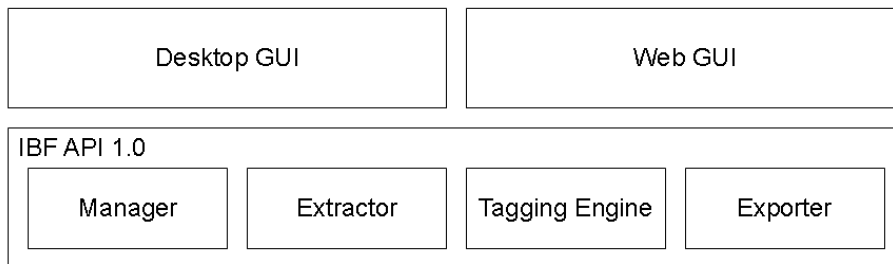


Fig. 3. IBF API 1.0 infrastructure.

The main research work was concentrated on the Book4All tool. In this paragraph we discuss its architecture, what issues occurred and which solutions we proposed during the design and the implementation phases.

The Book4All tool is divided into four main components:

1. Extractor;
2. Tagging Engine;
3. IBF Editor;
4. Exporter.

### **5.1 Extractor**

The Extractor is the component of the tool that extracts text and images. The extracted text contains raw information about fonts (size and face). The extracted images are stored in a directory related to each page. A heuristic engine tries to relocate the images within the pages. The heuristic algorithm is based on the knowledge that in the raw text there are groups of blank lines instead of images. The number of blank lines is proportional to the image size. Using these data, the heuristic engine identifies the image position by computing the consecutive blank lines.

### **5.2 Tagging Engine**

The Tagging engine marks the extracted raw text using the operator input. The operator associates tags with font size and face. Using the ad-hoc PDF viewer, the operator inspects the PDF to gather information about font, by simply selecting the interesting text. The viewer then returns font face and size.

The Tagging Engine marks all text elements with IBF tags and using the operator input, and tries to reconstruct ordered and unordered lists by using a heuristic. The Tagging Engine output is valid IBF, which is ready to be adapted and extended by the operator.

### **5.3 IBF Editor**

Book4All provides a powerful IBF editor that simplifies:

- Heading level editing;
- Paragraph editing;
- Addition and editing of image descriptions;
- Table reconstruction;
- Lists optimization and insertion;
- Note handling and editing.

#### 5.4 Exporter

The Exporter is the element that exports IBF file into final format such as XHTML 1.0 Strict or DAISY 3.0, by mainly using XSLT transformations. The flexibility of the Exporter is closely linked to the flexibility of XSLT: the new final format can be added by simply adding a new XSL Transformation. The final format XHTML 1.0 can be read through a common browser and a screen reader like Jaws<sup>2</sup>. The DAISY format can be read via a DAISY SW or HW reader. For example, **Errore. L'origine riferimento non è stata trovata.** shows the DAISY content obtained by extracting the book [18] by using Book4All. The DAISY version extracted by Book4All can be transferred to the symbian-based smartphone Nokia N95. A DAISY multimedia player<sup>3</sup> can be installed on the mobile phone. The DAISY player includes a voice synthesizer, which reads the XML- DAISY book without needing an additional screen reader loaded on the smartphone.



**Fig. 4.** Exporting in DAISY 3.0. Reading book index using a mobile screen reader.



**Fig. 5.** Exporting in DAISY 3.0. Reading book content using a mobile screen reader.

<sup>2</sup> Jaws for Windows, <http://www.freedomsscientific.com/products/fs/jaws-product-page.asp>

<sup>3</sup> Daisy player for Nokia N95 8 GB,

[http://codefactory.es/descargas/family\\_3/product\\_16/version\\_57/MobileDaisyPlayer\\_S60\\_3\\_0\\_v\\_2\\_2\\_3.zip](http://codefactory.es/descargas/family_3/product_16/version_57/MobileDaisyPlayer_S60_3_0_v_2_2_3.zip)

## 6 Conclusions and future work

As described in this paper, the existing manual process for conversion and adaptation of e-documents to an accessible and usable version for students with visual impairments is time consuming due to the numerous manipulations carried out manually by operators. With our approach several tasks can be performed in a semi-automatic way which makes the process easier and in addition saves time. The proposed tool is able to convert the source PDF document into more accessible DAISY and X/HTML versions. In this perspective, the proposed tool is also appropriate for preparing X/HTML documents to be published on the Web based on a PDF version. This can be useful for Public Administration that cannot rely on experts in developing accessible Web pages. Moreover, the tool could be also valuable for additional purposes, such as for generating a variety of output versions (e.g. Braille or audio formats). In short, from a single source version it is easy to obtain various accessible outputs. Future work will concern the development of features and components aimed to simplify the editing tasks as well as a Web-based usage in a collaborative way. For this purpose, tools like Microsoft Word and OpenOffice.org Writer plug-ins; a web application for conversion, adaptation, collaboration and sharing will be considered.

## Acknowledgements

This project is supported by the Italian Department of the Education.

## References

1. Adjouadi, M., Ruiz, E., Wang L. (2006). Automated Book Reader for Persons with Blindness. Proc. ICCHP'06, Linz, Austria, 2006, pp. 1094-1101.
2. Bottoni, P., Ferri, F., Grifoni, P., Marcante, A., Mussio, P., Padula, M., Reggiori, A. (2009). E-Document management in situated interactivity: the WIL approach. Universal Access in the Information Society, Vol. 8 (3), August, 2009, PP. 137-153.
3. Brzoza, P., Spinczyk D. (2006). Multimedia Browser for Internet Online Daisy Books. Proc. ICCHP'06, Linz, Austria, 2006, pp. 1087-1093.
4. Burgstahler, S. (2002). Universal Design of Distance Learning, Information Technology and Disabilities, <http://www.rit.edu/~easi/itd/itdv08n1/burgstahler.htm>.
5. Chen G., Li Q., Jia W. (2005). Automatically Generating an E-textbook on the Web, World Wide Web, Vol. 8, Issue 4, Dec 2005, Pp. 377 - 394.
6. Contini E., Leporini B., Paternò F. (2008). A Semi-automatic Support to Adapt E-Documents in an Accessible and Usable Format for Vision Impaired Users. Proc. of ICCHP'08, Linz, Austria, 2008, LNCS vol. 5105/2008, pp. 242-249.
7. DAISY/NISO Standard (2008). Structure Guidelines, at <http://www.DAISY.org/z3986/structure/>

8. Edmonds, C. (2004). Providing access to students with disabilities in online distance education: Legal and technical concerns for higher education. *American Journal of Distance Education*, 18(1).
9. Finnish Government Information Society, Ministry of Agriculture and Forestry, at [http://www.mmm.fi/attachments/maanmittausjapaikkatiedot/5g7g69SqF/Kansallinen\\_pai\\_kkatiestostrategia\\_Eng\\_2005\\_2010%5B1.pdf](http://www.mmm.fi/attachments/maanmittausjapaikkatiedot/5g7g69SqF/Kansallinen_pai_kkatiestostrategia_Eng_2005_2010%5B1.pdf)
10. Gil-Rodríguez, E. P., Planella-Ribera, J. (2008). Educational Uses of the e-Book: An Experience in a Virtual University Context. LNCS, Vol. 5298/2008, HCI and Usability for Education and Work, Springer Berlin, pp. 55-62.
11. Hardy, M. R. B., Brailsford, D. F., Thomas, P. T. (2004). Creating structured PDF files using XML templates. Proc. of the 2004 ACM symposium DocEng'04.
12. Hassan, T. (2009). Object-Level Document Analysis of PDF Files, Proc. Of ACM Symposium DocEng'09, Munich, Germany.
13. Hersh, M., Leporini, B. (2008). Making Conference CDs Accessible: A Practical Example. Proc. of ICCHP'08, Linz, Austria, LNCS vol. 5105/2008, pp. 326-333.
14. Kanahori, T., Suzuki, M. (2006). Scientific PDF Document Reader with Simple Interface for Visually Impaired People. LNCS, Vol. 4061/2006, pp. 48-52.
15. Leporini, B., Paternò, F. (2004). Increasing Usability when Interacting through Screen Readers. *Universal Access in the Information Society*, Vol. 3(1), 2004, Guidelines, Standards, Methods and Processes for Software Accessibility, pp. 57-70
16. NIMAS, National Instructional Materials Accessibility Standard, <http://nimas.cast.org/about/nimas>
17. Sun Y., Harper D. J., Watt S. N. K. (2004). Design of an e-book user interface and visualizations to support reading for comprehension. Proc. of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2004, pp.510-511.
18. Twitter Tips, Tricks, and Tweets - Paul McFedries. Pdf twitter: <http://mcfedries.com/cs/content/TwitterTTT.aspx>