Adding semantics in web-based digital libraries to support information seeking of blind people

MICHAIL SALAMPASIS
Department of Informatics,
T.E.I. of Thessaloniki
Thessaloniki, 57 400, Greece
+30 2310 791284

CHR. KOUROUPETROGLOU
Department of Informatics,
T.E.I. of Thessaloniki
Thessaloniki, 57 400, Greece
+30 2310 791604

DIMITRIS TEKTONIDIS
ALTEC S.A., Research
Programmes Division,
Thessaloniki, Greece,
+30 2310 595659

Abstract - : Universal accessibility is one of the primary targets in the development of the World Wide Web (WWW), but unfortunately some groups of people have problems in accessing the WWW efficiently. Clearly, blind users are such a group that must deal firstly with the reading task of a web page. Secondly, because the WWW has turned into a highly interactive media and web pages are designed with sighted users in mind, blind users have to overcome the problem of inefficient within-web-page and across document browsing. This paper presents and discusses an ongoing research effort which aims to exploit the idea of semantic web for improving the efficiency and effectiveness of the information seeking process of blind users. A set of tools are described which are used to semantically enrich web-based digital libraries and therefore make browsing of blind people more rational and therefore more efficient. Taken together the proposed tools (an ontology editor, an annotation tool and a specialized voice web browser) shape a methodology that could produce semantically enhanced web-based digital libraries, and could potentially make information seeking in the WWW easier for the blind people.

Keywords: Semantic web, semantically enhanced digital libraries, voice web browser, annotated web pages.

1 Introduction

Internet and the World Wide Web (WWW) especially are nowadays a vast information and knowledge source. All this information and knowledge is used for a wide range of daily activities such as question answering, problem solving, education, work, entertainment, using public or administrative services etc. Unfortunately, navigation of this complex hypermedia environment and information seeking in this vast information source is not an easy task. The problems of high cognitive overhead, user disorientation, low efficiency in such large electronic information seeking environments, has long been considered as serious issues in the relevant literature [10].

The potentially complex task of information seeking in the WWW is further complicated when the end-user is blind or visually impaired (VI). The development of Graphical User Interfaces and windows based environments made this problem even more difficult because the WWW followed the same graphical design paradigm making the information seeking task difficult for blind people. It is eminent, that web pages are designed with sighted users in mind which are experienced in the desktop metaphor. A typical web page presents a lot of information which is visually fragmented and organized into groups. Also, navigational aids such as navigation bars or menus are designed and located in the screen having sighted users in mind. User studies show that sighted users can orient themselves within this complex page design within few seconds [3].

Unfortunately, for blind people or visually impaired all these visual cues and navigation aids are lost. Typical screen readers used by VI read the information in the web page from the top left to the bottom right, without providing any assistance to the VI to comprehend the structure of the web page under examination or to move to a particular web page element or navigational aid.

To alleviate the accessibility problem for VI, the last few years, some ideas have come forward from initiatives such as the W3C Web Accessibility Initiative (WAI) and several ideas have been suggested. More specific the W3C WAI has published a set of guidelines referring to web authors [2], and also to developers of web browsers [7] and web authoring tools [12], which provide a common standard for assessing and improving the accessibility of web content, agents and authoring tools. However, the W3C guidelines are mainly concerned with text alternatives, correct use of markup. An overall systematic analysis of how VI users browse and seek information in the WWW is missing. Our research work is mainly driven by our belief that despite the ground vision set by the W3C accessibility guidelines, complementary work in other directions is required on top of the work done by the W3C WAI to deal with the accessibility problem in a more complete and effective manner. In our opinion, these complementary directions should carefully examine the WWW as a complex and highly interactive digital library, hypermedia-based information seeking environment. If the WWW is examined using this approach, the issues and problems that confront VI in efficient browsing or effective seeking information in the WWW could be dealt better.

For example, apart from the accessibility problems discussed in W3C's guidelines, there is also another parameter that today emerges as new aspect of the problem. This is the usability of accessible web pages and in general how seeking information, could be less disorientated and develop less cognitive overhead for blind users. This simply means that regardless of a web page is rated as accessible according to the existing W3C rules, there might still be some problems for blind users, to retrieve the information required. Globe et al [3][4], describe these kind of problems that VI confront when navigating in the WWW using the metaphor of traveling in an environment such as a city. It is clear cut, usability problems are even more difficult than accessibility ones. Accessibility problems have to do more with the syntactic aspects of a web page. On the other hand the usability issues occur during the complex information seeking process.

For accessibility problems W3C WAI has suggested an approach that is based on the idea of "Design for all" [11]. Web sites developed based on this idea separate the actual content from the presentation of it. The main benefit from this approach is that different representations may be produced for the same information. To this direction, there are today standards such as the Cascading Style Sheets (CSS). CSS files gather a set of presentation rules that are applied in a web page. CSS leave the web page only with its structural and semantic elements such as headings, tables, lists etc. The accessibility features that are provided from CSS are also analyzed by W3C [7]. In addition to the standard rules of CSS the last W3C recommendation on CSS level 2 revision 1, includes a feature of Aural Style Sheets for aural representation of documents. This means that a developer who uses this module can define the changes in the voice that are needed and improve this way the aural presentation of a document.

In addition to the separation of structural and representational elements in web pages there is also the concept of metadata within web pages that is widely used in the "Design for all" principles. Metadata are information, in a machine readable form, that describes the content of a document. Metadata is a key concept to the notion of the Semantic Web [1]. In Semantic Web content is not only human-understandable but machine-understandable too. This means that programs such as web browsers or web crawlers will be able to understand the meaning of documents and process them appropriately. In our research problem, metadata could be used for many reasons such as to describe to the blind user what kind of information exists on a page and what kind of visual cues or navigational aids are available to assist him in the navigation of the document.

Technically the use of metadata was actually implemented and boosted by the development of eXtensible Mark-up Language (XML) that is for the creation of customized languages for different purposes. Today they are already known many applications of XML for a wide range of purposes such as MathML, SVG etc. In the field of web accessibility, W3C has lately proposed as recommendation a new language especially for aural presentation of document called Speech Synthesis Markup Language (SSML). Another language developed for similar reason is VoiceXML which helps in producing voice controlled web applications.

XML also inspired researchers who proposed solutions for accessibility and usability problems. Huang and Sundaresan [5] [6] for example, use it for transforming web pages so that they meet user needs when dealing with service providing web sites. Yesilada uses it [14] in order to improve mobility and usability issues of blind users in the web.

The latter show that metadata and XML as the new trend of the Semantic Web can become very helpful in the solution of many issues. In our work metadata and the general idea of Semantic Web is applied to solve usability issues in the web and improve the browsing experience of blind users. More specifically, we aim in making their information seeking process in the web more effective and efficient. The goal is to enable blind users navigate the web in a more rationalized way, closer to the one used by sighted users.

2 Adding and using semantics

It is true the idea of Semantic Web has raised a series of different expectations. As Marshal and Shipman state [15] “The Semantic Web is the outgrowth of many diverse desires and influences, all aimed at making better use of the Web as it stands.” From the W3C’s inception, there was a perceived need to bring order to the loosely connected networks of digital documents that made up the Web. Although this order was to be realized by consortium’s development of standards, it would also reflect the order that libraries have and the Web does not – a consistent structure by which people can access materials.

In our research work the idea of Semantic Web is used for two specific purposes. The first is to provide web authors a method to make navigation of their web pages easier for blind people by allowing easier access of blind users in navigational aids. It is well known that in order to support across document browsing usually designers and developers of web applications provide users with various navigational aids such as navigational bars, guided tours or landmark documents that could assist them in the information seeking process. Blind users need these aids even more, because this is how they navigate in physical environments. However, the detection, recognition and approach of these aids is mostly based on their visual appearance (e.g. color, position on the page). This design prevents blind users to use these navigational aids because they cannot identify them easy.

It should be stressed that the issues raised and the problems discussed in the previous paragraph are not solved by guidelines such as the W3C Accessibility Guidelines (WCAG). WCAG is a set of rules for overcoming accessibility problems of VI, in an effort to ensure that web pages and user interface elements could be reasonably accessed by VI without extreme difficulty and perhaps without discarding any information. However, these guidelines deal more with accessibility issues of user interface elements, and do not take into strong consideration the problems encountered during an information seeking process that have been discussed previously.

In our work, metadata are firstly used to semantically enrich web pages for navigational purposes. One method to enrich web pages is to identify the various elements such as menus, search boxes, site maps etc. which are available in web pages. Metadata can be utilized in various ways. The first one is to supply the blind user with browsing shortcuts to this visual cues and navigational aids. This gives blind users a variety of landmarks within a page to make better the within page browsing process.

The second use of metadata in our work is to semantically annotate the actual content of web pages. This type of annotation can be utilized in various ways too. The first one is similar to browsing shortcuts for visual cues and navigational aids (e.g. move directly to sports news headlines on a portal page). Similarly, web pages could also be rearranged from the browser based on either previous user behavior or user request. In addition this information will mainly help in the across document browsing since it will enable a scanning simulation process which will give a brief overview of the page content (e.g. this page has 3 sports headlines about football and 5 about basketball).

From a technical perspective, the framework which is discussed above requires three components/software tools which in our implementation are:

- a specialized ontology editor called *ONAR* which is used to create *a semantic vocabulary* based on an ontological approach.
- An annotation tool, which is used by web authors or specially trained annotators to enrich web pages with annotations which are integrated internally into web pages or stored externally in annotation files using RDF form.
- A specialized voice web browser, called *Polephemos*, which exploits the annotations accompanied a web page in order to semantically enhance the browsing.

2.1 ONAR – an Ontology Editor

The tool which is developed to create the vocabulary used for annotations is an ontology editor called ONAR. Using ONAR annotators use a vocabulary (described using OWL) containing definitions for visual elements or navigational aids commonly used in web sites, such as menus, table of contents, site maps, navigation bars etc.

The utilization of OWL in order to define semantics is based on a graphical environment that enables the web designer to create ontologies that will be used for the annotation. The Conceptualization Designer (Figure 1) follows the frame based logic of Web ontologies.

The Conceptualization Designer enables the web designer to graphically define the entities (OWL classes) and their relations (OWL object properties) using a reduced, simplified set of OWL language. The set of the OWL language elements that the Conceptualization Designer contains the following elements:

a) *Classes:* OWL classes and inheritance is supported. The designer can define entities of instance based on OWL standard.
b) *Object properties:* Object properties are supported in order the tool to facilitate the semantic relations between concepts
c) *Datatype Properties:* The designer can define special characteristics of classes as Datatype properties. Also the tool enable cardinality as it is defined in OWL-Lite.

The composition of the ontologies is done using the graphical environment while the tool produces the corresponding OWL file. The tool enables the import of ontologies in order to facilitate the composition of complex ontologies from a set of ontologies.

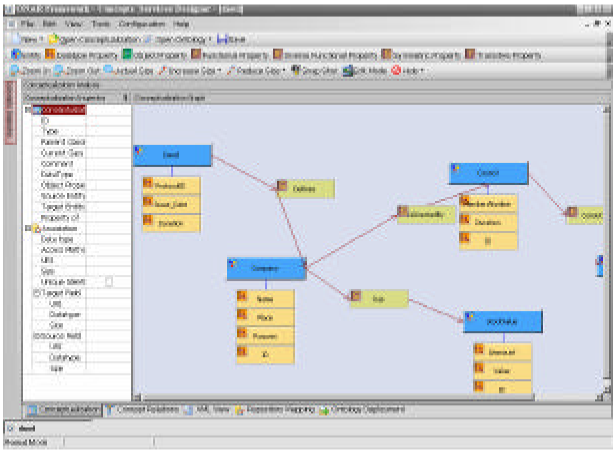

Figure 1: The conceptualization designer

2.2 Annotation tool

The metadata which are used by the voice web browser (Polephemos) are integrated into web pages using an annotation tool. The annotation tool enables web authors and annotators to produce annotations without needing to know their actual syntactic form. The set of annotations for a particular web page can be stored in two possible ways. The first is an HTML or XHTML file containing both the web page content and the additional metadata needed. The second is to an external Resource Description Framework (RDF) file containing the annotations for a web page or a list of web pages (based on a URL pattern matching algorithm). This type of external annotation file is also used for creating annotation templates for a set of web pages by describing their URL using a regular expression. This was done in order to be able to annotate a whole web site or a subset, if it follows the same structure, based on the annotation of a single web page.

2. 3 Polephemos

The third tool in the proposed application framework is a specialized voice web browser called Polephemos. The main idea behind Polephemos is to have a browser that exploits metadata and provide useful information about visual cues, structural elements and navigational aids to the blind user.

At the current stage of our research work the vocabulary was focused on structural elements and visual cues that exist in web pages and would be helpful for blind users to know their existence and browse amongst them. The Polephemos browser also focuses currently on providing solutions for more efficient within-web-page browsing. However, the principles used are applicable in order to use them for across-document browsing. Finally separately from this basic idea, there are also a number of other features in the Polephemos prototype that facilitate basic browsing within web pages.

2.3.1 Shortcuts selection

The selection of keyboard shortcuts for the use of the browser is a very crucial point when having to do with blind users that have to learn and remember them. Many of the browser's functions are inevitably common with Internet Explorer (IE) so the shortcuts selected for these functionalities were the same with those used in IE. This way the learning curve of Polephemos could be shorter. For the selection of other features' shortcuts, not present in IE, Jaws which is a widely used screen reader was considered as an inspiration. This decision was made mainly because of the background of the users that were going to test the browser in the experiment described further down in the article. Finally for all other functions not included in either of the previous tools (IE and Jaws) the shortcuts were based on initial letters of the functions. This would make easier their memorization.

2.3.2 Basic web browsing functions.

Polephemos provides a series of basic browsing functions that facilitate general purpose browsing in the WWW. First of all it has the ability, as all browsers, to move back and forward in web pages already visited using the Alt and Left of Right arrows respectively. It can also move directly to the home page defined in the system by pressing Alt+Home. The user can enter his own address by using either Alt+D or Ins+F7 shortcut and can read the URL currently browsing be pressing Alt+R. Finally he can move directly to the beginning or the end of a web page using the Home and End keys respectively and can find certain text within the document using Control+F to type in text to be searched and F3 on order to find more instances of the text within the document.

Polephemos analyses a web page using the Document Object Model (DOM) of an HTML document. The analysis of a document based on DOM is done in two levels. The first one returns a list of HTML elements to be heard by the blind user, called browsing list. The browsing list depends on the browsing mode selected (full content or links only are implemented so far). The full content browsing mode (selected pressing Alt+A) contains elements that when added together, don't subtract any information from the whole document.

As already said, the browsing list changes when the browsing mode changes. Browsing mode is the first of the two levels of decomposition of the document. In this first prototype apart from the browsing mode to the full content there is also a browsing mode for browsing the links of a web page. Changing the browsing mode affects the browsing list and changes it so that it contains the appropriate elements. In the case of links mode (selected pressing Alt+L) the list consists of all the links included in a web page.

2.3.3 Speaking modes

The second level of decomposition is the one that combines the HTML and DOM analysis with the text-to-speech (TtS) engine. In Polephemos the speech is produced by DEMOSTHeNES Speech Composer which is a general-purpose multilingual and polyglot software (TtS) system that supports the English and the Greek languages [13]. The user has 3 options in selecting a speaking mode, i.e paragraphs, sentences or words.

2.3.4 Special elements browsing

The distinctive feature of Polephemos is that it is able to identify certain annotations within an HTML document and use them to inform the user about the existence of various special elements such as navigational aids, menus, forms etc. The user hears the list of the browsing shortcuts, and can select one to move his reading focus to this point immediately. At this first development stage, Polephemos is able to identify these annotations by having a list of Attribute Names, Attribute Values and Descriptions. This means that it searches for HTML elements having specific "reserved" attribute names and values and if found it uses the respective description to inform the user that the specific visual element exists. This list is kept in a comma separated text file that contains triples of the previous variables. This set of triples is formed based on the vocabulary that will be described further down.

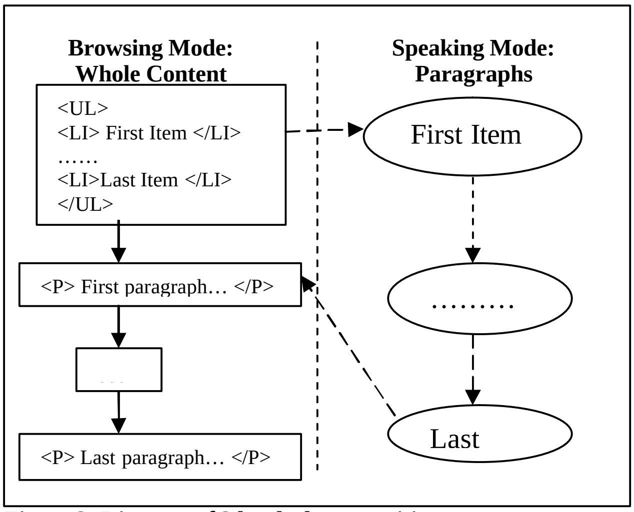

Figure 2: Diagram of 2 levels decomposition

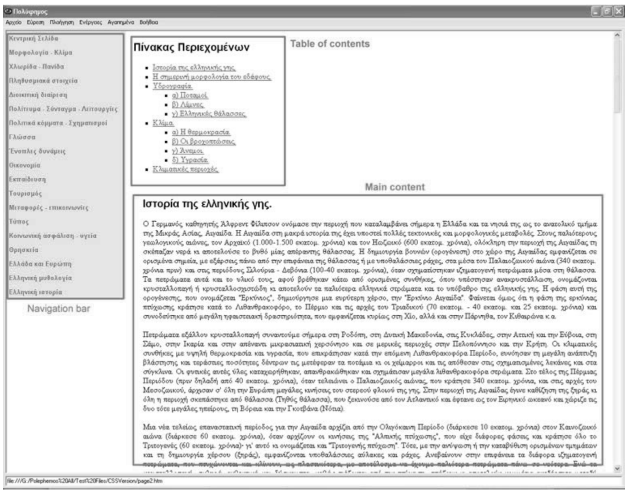

Figure 3 : Screenshot of Polephemos with available special elements shown in rectangles

Web pages that have elements with these certain Attributes and their Values can be recognized by the Polephemos browser which will give the user even more functionality. The browser informs the user of their existence and enables him in moving directly to each of them. As an example let's consider a web page constructed having accessibility rules in mind and using CSS for the presentation of various elements on a web page. If the following code is used for a menu, then the triple needed in the text file previously described is "id, menu, Navigation Menu". Since this triple exists, Polephemos will identify and inform the user that there is a menu on the web page.

```
<div id="menu">
<ul>
<li> Home Page</li>
<li> Events</li>
<li> Links</li>
</ul>
</div>
```

When the user wants to identify possible elements that exist in a web page uses the Alt+I shortcut and the browser will then inform him if any annotations-elements were found. The elements found are automatically placed on a list that can be accessed in a circular way using the Alt+Up and Down shortcuts. Every time the user moves to an element the browser informs where s/he is and can then browse the page by using the Up and Down keys. When the user reaches the end of the element he selected, s/he can continue browsing to the rest of the page as if he was reading the whole text.

3 A case study – adding semantics to a web-based encyclopedia

To apply our methodology and to construct an experimental environment for evaluation purposes, we have used as a testbed the web based Greek encyclopedia "Science & Life" (www.gnosinet.gr). The encyclopedia is the largest Greek on-line encyclopedia with over 150.000 text and lexicon articles, and about 15.000 multimedia items (maps, photographs etc). It provides multiple ways for information seeking (thematic, alphabetic, query-based, multimedia retrieval) and it is a quite demanding information seeking environment.

In our previous visits in Blind people's schools to discuss their problems in seeking information on the WWW, we have noticed the difficulty that blind users had to use the on-line encyclopedia. Therefore, we decided to apply our methodology in this web-based digital library.

We have selected a subset from encyclopedia's content and we have created a vocabulary using ONAR to describe the various elements of encyclopedia's web pages. A web site was constructed having in mind accessibility and also containing metadata in the form that was described in Section 2. The metadata were integrated to the web pages by using the annotation tool and the vocabulary described above. To keep the experimental environment simple, only 3 special elements were available: Navigation bar, table of contents and main content (Figure 3).

An initial experiment was conducted using six blind users. The 6 blind users involved were separated in 2 groups. Each user conducted 3 search sessions based on a given question. The first group was given one set of questions to answer and users were not informed for the existence of semantically enhanced browsing. In the second group users were informed of the semantically enhanced browsing and how to use it.

Although some marginal differences were measured, statistical analysis of the log files showed no significant differences in terms of time and keystrokes needed for answering a question. However, we believe that a larger experimental environment with more shortcuts available and a bigger set of information could give important differences.

The questionnaire that was answered by blind users after the experiment, provided useful feedback indicating that the browser was quite user friendly and the shortcuts feature was very helpful. More specific all users rated as highly useful the shortcuts feature and the usability increased radically when the feature was used.

Finally, in the interviews after the experiment most of the users pointed out the usefulness of the shortcuts and asked for further development of this feature. This is quite encouraging to continue our work in this direction.

In the time of writing this paper we are preparing a larger experiment using the complete web based encyclopedia and all the features and information seeking methods which are available in it. We aim to evaluate and compare the information seeking performance of blind users with and without Polephemos (semantically enhanced browsing in comparison to "normal" browsing). We also aim to compare the performance of sighted users with this of blind users using semantically enhanced browsing.

4 Limitations & Conclusions

The paper presented the idea of exploiting metadata within web pages in the implementation of a web browser for blind users that would enable them to browse quicker and easier within web pages. The

"Semantic Web" which is the new trend in web development can also become useful in cases such as this. From the initial evaluation of the prototype the results showed a good response of the blind users to the idea and would like to see it implemented even further.

Of course, conducting user-centered experiments is an extremely difficult task and usually bounded with limitations. First, because there are so many different variables that someone must take into account. Second, it requires a large effort to organise a user-centered evaluation. The experiment which is briefly described in this paper is large enough (6 blind users were involved conducting 36 search sessions), but clearly further experiments are required to reach statistically significant conclusions. Because the results cannot be validated statistically, the views and statements reported in this paper should be regarded as indicative and tentative.

To expand this idea there are two major issues that should be taken into account. First of all, not all web pages can include these annotations since it is impossible to enforce web developers use those special characteristic. Here comes the power of "Semantic Web" and external metadata. Those metadata could be exported in separate files and be used in order to make possible the feature. This leads also to the problem of where and how metadata will be stored in order to add flexibility and communication of various annotators. Some interesting solutions have to do with RDF and RDF databases such as those used in the Annotea [9] project. The idea there is to have a network of RDF databases that will store these metadata and provide it to the users when requested. Annotea can be a good example as it is a platform that uses already widely used standards in order to work. This means that there is no need for learning and adapting to new technologies and standards and this is an advantage for web servers and their administrators.

The second major issue is the vocabulary that will be used to define elements having browsing shortcuts. This vocabulary is currently under development as already said previously from a team of researchers working on that project and at its first stage includes visual elements often used in web pages such as search form, menus, main context etc. This vocabulary could be further developed or blended with other vocabularies concerning various domains such as educational sites so that it would help not only in the browsing within a certain page but across a collection of pages such as the whole web too.

The previous two aspects lead also to the problem of metadata creation. This area has mostly to do with web site development and a specific tool is also being developed and was used in order to create the metadata within the web pages of the experiment. This tool provides the designers and other people with an environment that produces this metadata automatically by using a visual interface easy to be used by any user that has some basic knowledge in HTML. Using this tool, annotators could produce external annotation files that could help blind users of Polephemos. At this first stage of it, there is also an element of pattern matching included by providing the feature of creating an annotation template for a set of web pages based on regular expressions used within URLs. In addition to it further heuristics can be developed and added to it in order to facilitate easier, quicker and semi-automatic annotation of web sites.

Finally the ultimate object is to make this metadata created automatically using pattern matching algorithms and information already included within web pages that many authoring tools produce. Further research in fields of artificial intelligence and language processing can be really helpful in providing even more advanced heuristics that will finally enable the annotation of web sites in an almost automatic way. Until then annotators and web authors will have to keep an eye at these annotation systems and cooperate in order to have reliable and well functioning systems.

ACKNOWLEDGMENTS

The research project is funded by the E.U. and the Greek Ministry of Education under the research program "Archimedes". We would also like to thank Dennis Roussel for his help in the annotation tool development.

5 REFERENCES

1. Berners-Lee, T. Semantic Web roadmap. Available at http://www.w3c.org/DesignIssues/Semantic
2. Chisholm, W., Vanderheiden, G. and Jacobs, I. Web Content Accessibility Guidelines 1.0 Available at http://www.w3.org/TR/WCAG10/wai-pageauth.html
3. Goble, C., Harper, S. and Stevens, R. The travails of visually impaired web travelers, In: *Proceedings of the eleventh ACM on Hypertext and hypermedia*, ACM Press, 2000, 1-10.

4. Harper, S., Goble, C. and Stevens, R. A pilot study to examine the mobility problems of visually impaired users travelling the web, *SIGCAPH Comput. Phys. Handicap.*, 2000, (68) 10-19.
5. Huang, A. W. and Sundaresan, N. A semantic transcoding system to adapt Web services for users with disabilities, *Proceedings of the fourth international ACM conference on Assistive technologies*, 2000, 156-163.
6. Huang, A. and Sundaresan, N. Aurora: a conceptual model for Web-content adaptation to support the universal usability of Web-based services, *Proceedings on the 2000 conference on Universal Usability* , 2000, 124-131.
7. Jacobs, I., Gunderson, J. and Hansen, E. *User Agent Accessibility Guidelines 1.0* Available at http://www.w3.org/TR/UAAG10/cover.html
8. Jacobs, I. and Brewer, J. *Accessibility Features of CSS* Available at http://www.w3.org/TR/CSS-access
9. Kahan, J. and Koivunen, M.-R. Annotea: an open RDF infrastructure for shared Web annotations, *WWW '01: Proceedings of the tenth international conference on World Wide Web*, 2001, 623-632.
10. Marchionini, G. *Information Seeking in Electronic Environments*, Cambridge University Press, Cambridge, UK, 1995.
11. Savidis, A., & Stephanidis, C. Unified User Interface Design: Designing Universally Accessible Interactions. *International Journal of Interacting with Computers*, 2004, 16 (2), 243-270.
12. Treviranus, J., McCathieNevile, C., Jacobs, I. and Richards, J. *Authoring Tool Accessibility Guidelines 1.0* Available at http://www.w3.org/TR/ATAG10/atag10.html
13. Xydas G. and Kouroupetroglou G. The DEMOSTHeNES Speech Composer, Proceedings of the 4th ISCA Tutorial and Workshop on Speech Synthesis (SSW4), Perthshire, Scotland, September 2001, pp. 167-172.
14. Yesilada, Y., Stevens, R. and Goble, C. A foundation for tool based mobility support for visually impaired web users, Proceedings of the twelfth international conference on World Wide Web, 2003, 422-430.
15. Marshal C., Shipman F. Which Semantic Web? In proceedings of the ACM HT03 conference, August 26-30, 2003.