CYNTHIA : An HTML Browser for Visually Handicapped People

Mathieu Raynal and Mathieu Serrurier

Daumas Informatique SA,
Paris Cyber-Village
204, rue de Crimée
75019 Paris, France
{raynal,serrurie}@irit.fr
http://www.daumas-informatique.fr

Abstract. This paper deals with an HTML browser for visually impaired people, named CYNTHIA. After a short description of the principle and techniques of interaction, we present the two conception approaches of the intern structure of this browser : the first one integrates VOICEXML and a structure for the visual information. The other one is based on a chained structure allowing both the interaction and the presentation of information. We compare these two approches and give the results in terms of conception and time.

1 Introduction

Nowadays, a lot of documents are available on the Web. This wide range of information should theoretically be accessible for everyone. Although many studies were done, resulting notably in the WAI's recommandations [1], very few HTML documents follows these guidelines.

Currently two possibilities are offered to visually impaired people to have access to Internet : either they use a commercial navigator (Internet Explorer, Netscape ...) coupled with a screen reader such as JAWS [2] or VIRGO [3] or they use specialized navigators such as HOMEPAGE READER [4], PW-WEBSPEAK [5] or BRAILLESURF [6] for instance.

INOVA[7] studies which had led to the design of BRAILLESURF, allowed us to know the both blind persons' and partially sighted persons' needs for presentation. We aim at testing the contribution of new technologies to these specialized browsers. Moreover, for the visually impaired people, the vocal interaction is one of the main ways to present information. VOICEXML[8], which is a XML-based markup language used for the spoken man-machine dialogue, seems to be appropriate to convert HTML file and its graphical presentation into a vocal presentation.

This paper firstly presents our HTML browser CYNTHIA. We will then discuss two design approches used for CYNTHIA development, one using VOICEXML and the other including a chained struture. We will finally compare those two methods by underlining their advantages and limitations of design and efficiency.

We aim at defining the efficiency of VoiceXML for the interaction through Html documents for blind and visually impaired people.

2 CYNTHIA Description

2.1 Principles Retained for CYNTHIA from INOVA Studies

Cynthia is based on the principles already tested through the Braillesurf[9] navigator. We have integrated technologies which appeared to be able to improve portability (thanks to Java language) and interfacing (with SAPI[1][10] interface for the output). Inova studies focus on blind person's and partially sighted persons' needs (fontsize, color ...). According to these ones, we created our user interface, with a specific presentation of information.

Before approaching the description of Cynthia's internal structure, we are going to introduce our system specifications.

2.2 The Techniques of Interaction

Cynthia's architecture allows to add new input and output modalities. Actually, to interact with the browser, the user can only use the keyboard. If interaction was only limited to browsing the Html document, we could have used spoken commands, as these ones have already proved in the past their efficiency [11]. But the problem of the misrecognition in fields input remains. In order to avoid these errors, we decided to focus on the keyboard as the only input modality available for the moment.

The user interface combines the vocal and visual modalities. The visual modality enables the partially sighted to have a persistent access to the information. The vocal modality aims at giving the blind people the possibility to acquire the same information. The synthetized text is in parallel highlighted on the top of the graphical interface allowing to use both vocal and visual modalities simultaneously. And this allows the visually impaired persons to be able to work more easily with sighted persons.

2.3 Adaptability : a CYNTHIA's Property

With our browser, we give the user the possibility to configure its interface according to his needs. Once the configuration is realized, all the documents will be presented according to these specifications which are saved by the browser. Our hypothesis is that a user could have difficulties with certain configurations (like a black font on a white background). It is for this reason that neither the Css[2], nor their extension Acss[3] defined by the W3C are taken into account. Only the name of Html tags and the arguments they contain are interpreted to present document information.

[1] Speech Application Programming Interface
[2] Cascading Style Sheet
[3] Aural Cascading Style Sheet

2.4 The Functioning Principle

The functioning and navigation principles in CYNTHIA are the following one : after having recovered the HTML file on INTERNET, the purpose is to extract from it the textual information, to interpret those that are normally visual[4] and to restore the whole to the user via the dialogue manager.

For the whole part concerning the analysis and the conversion of the HTML file into an internal representation structure, we analyzed two approaches of conception (see Fig. 1). On the one hand, we realized this part with an integrating structure of VOICEXML. On the other hand, we realized a second version of our navigator with a chained structure representing the HTML document, allowing to compare and to estimate the possible contribution of VoiceXML to the navigator.

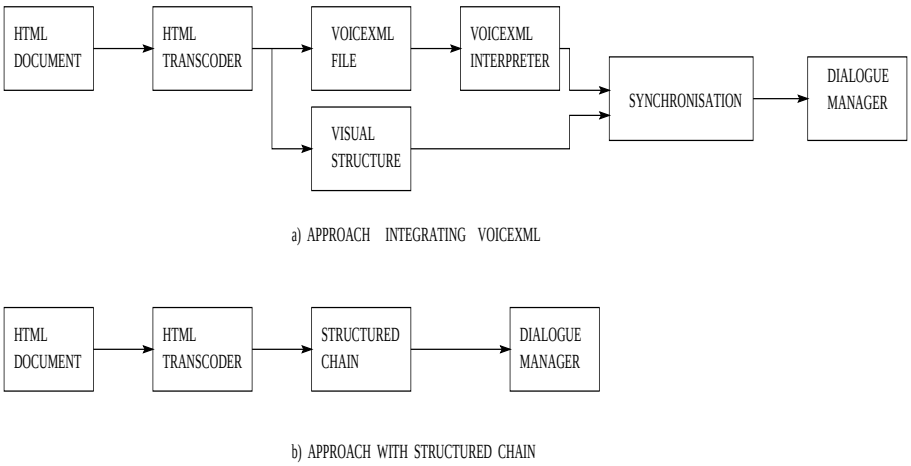


Fig. 1. Representation of the two conception approaches

3 Approach Integrating VOICEXML

3.1 VOICEXML Description

VOICEXML is a language published by W3C[12]. VOICEXML is based on XML principles : it brings an answer to all XML specifications, and offers description markers. This is one of its main advantage as many tools are available to create, transform and parse XML format [13].

[4] that is to say extract from tags the textual information which could replace the graphical representation, like the argument ALT of an IMG tag

VoiceXML is defined to design spoken man-machine system[5]. It offers to the user voice interaction technique based on Asr[6] and Tts[7]. This appeared to be an advantage as we wished to offer voice as main modality. Moreover, Html and VoiceXML are two markup languages. So, the Html conversion to VoiceXML seems to be relatively simple, for numerous concepts and tags.

Considering these main advantages, we realized a browser integrating an internal structure based on VoiceXML.

3.2 Structure Description VOICEXML

To integrate VoiceXML to our navigator, some preliminary tasks had to be completed before rendering to the user the HTML document information.

- on one hand, VoiceXML allows only vocal presentation of the information. To fulfill this lack of presentation, we realized a structure, called visualization structure, storing the necessary data to visualize the HTML document information;
- on the other hand, VoiceXML is an interpreted language. When receiving an HTML file, it must be converted into VoiceXML file. Once this preliminary step completed, the VoiceXML file must be interpreted, so that the user may interact with the file information.

To take into account these contingencies, the processing steps of Cynthia are :

- Html file recovery,
- Html file conversion to VoiceXML, with, in parallel, the visualization structure generation.

As soon as these steps are completed, a dialogue manager gives using commands to VoiceXML interpretor. It will be in charge with the information rendering to the user. Dialogue manager controls simultaneously the vocal output with the graphic interface.

4 Our Structural Approach

This method objective is to exploit the VoiceXML advantages and consider its contingencies. We therefore created a structured chain. Each node represents a part of an Html file, which can be either text or data located between an opening markup and a closing markup, or markup arguments which may give complementary information to the user.

On one hand, in each node, the information is stored twice : there is firtly information which is display on the screen, and second time for information

[5] VoiceXML is mainly designed for the conception of interactive voice system accessible by phone

[6] Automatic Speech Recognition

[7] Text To Speech

which is vocally synthetized. The information is stored for each output modality, because what will appear on screen and what will be announced by the synthesis may be slightly different.

As example, a link called "ICCHP website"
On a visual browser, the graphic interface will be :

ICCHP website

On CYNTHIA navigator, the graphic interface will be :

[ICCHP website]

and the synthesis will announce :

link ICCHP website

On the other hand, in each node, we have the possibility of allotting an action. This action represents the possible interaction with information contained in the node. Each type of interaction is represented by an attribute and / or a method in the node. If the interaction type is not mentioned for an information, the corresponding attributes will be null. Thanks to this process, we have a structure where nodes can represent any interaction type.

5 Pemilinary Comparison of these Both Approaches

The both approaches for this navigator realization are based on its representation structure of the HTML content and interaction mecanisms. Despite the differences between the structures, the user interface and his option remain the same. Information presentation and navigation into the document also remain similar from a structure to another.

5.1 VOICEXML Limits for this Type of Application

Although some functionalities are similar between HTML and VOICEXML, these two languages differ on many points :

- The VOICEXML specification lies on information presentation under the form of dialogue. The dialogue is described by a tree which defines the different ways of reading a document. In HTML, the way of reading a document is suggested by its graphical representation, which is not systematically explained in the code (as an example, the sense of reading a table is not obvious). It often happens that a web page must not be read in the sense, in which it has been encoded.
- In many cases, the conception of a web page does not allow to read it entirely. The target is to access to information the faster possible, and to have the possibility of reading it more closely later. It is very difficult to generate automatically an accurate synthesis for a document, encoded in HTML : it then appears necessary to have the possibility of jumping a part of the document,

of reviewing it or launching a research. But the VoiceXML specifications do not make it possible. If a research is launched, we may access to any part of a document, and this is impossible to be defined thanks to the dialogue tree.
- Some Html markups are rather difficult to translate into VoiceXML. For example, there are anchor markups in Html : the interaction is represented in the same way as a link, making possible to point up directly on a precise part of the document, and not systematically on its beginning. This action, coming regularly in Html documents, can not be realized with VoiceXML markups.

These differences make impossible to interpret in a standard way a VoiceXML document.

5.2 Performances Comparison

Tests processing Tests have been carried out on processing execution times for these both representation structures. We aimed at calculating the time required to present the content of an HTML document. This time was estimated from the file reception by the navigator to the moment of the information restitution to the user. The tests were performed with a 1Ghz computer with 112Mo Ram. These tests were carried out on 7 differently sized files, containing the most representative HTML documents markups. Each file was tested on both version navigators of Cynthia, the first including VoiceXML and the second without VoiceXML (see Fig. 1). For each file, and each navigator, tests were performed 5 times. The 5 test results were so slightly different, that average result was taken as reference.

Results analysis The experience times are mentioned in the Tab 1.

Table 1. The files sizes are indicated in ko. Times are indicated in milliseconds.

File size	Time VoiceXML	Time without VoiceXML
10	750	70
20	1200	150
50	2700	350
100	5500	800
200	13000	1550
500	35500	3500
1000	86000	8750

It appears that computing times are ten times superior with the navigator integrating VoiceXML. For explanation, the structure is more complex to

generate : as already mentioned in the structure description, the structure is consists of two parts. Even if the HTML file is converted into these two parts, we may think more time is required for this double conversion. But there is consequent difference between both structures : after the HTML file conversion, the VoiceXML structure has to interpret the VoiceXML file previously generated. This interpretation is only completed in this structure and represents a large analysis part. The second point is, whatever structure is concerned, times evolution according to the file size is linear.

5.3 Conclusion and Perspectives

When comparing these both approaches, VoiceXML structure appears not to be adapted for this type of application. Considering the language flexibility and the conversion process, the structural approach seems to be more efficient.

In the context of the European project Vickie[14], whose aim is the conception of an electronic schoolbag dedicated to visually impaired pupils, Cynthia will be useful as base work. In this case, the navigator's using will be enlarged to different documents based on other VoiceXML languages. With the aim of interpreting more complex documents, VoiceXML limits are too consequent to carry out the project with this structure. The structural approach appears to be the most adequate solution, as it is not based on a specific language and offers more flexibility.

5.4 Special Thanks

Special Thanks to Inova team, and more precisely Dominique Burger for the contribution of their experiment and their previous studies for Braillesurf realization.

Special Thanks to members of Diamant team[15], and particularly to Nadine Vigouroux and Régis Privat for their comments in the article redaction.

References

1. http://www.w3.org/WAI
2. Jaws For Windows (Henter-Joyce, www.hj.com)
3. Virgo (Baum, www.baum.fr)
4. HomePage Reader (IBM www.ibm.com)
5. PW-WebSpeak (ProductivityWorks www.prodworks.com)
6. BrailleSurf (BrailleNet www.braillenet.jussieu.fr)
7. http://snv.jussieu.fr/inova
8. http://www.voicexml.org
9. Burger, B., Hadjadj, D.: BrailleSurf : An HTML browser for visually handicapped people CSUN 99
10. http://snv.microsoft.com/speech
11. Oriola, B., Vigouroux, N., Dcoret, Cl. Voice Recognition and KeyBoard as Interaction Inputs for Blind People : Analysis of Behaviour Users ICCHP'96 Linz, 1996, p.731–739

12. http://www.w3.org
13. Lucas, B: VoiceXML for Web-Based Distributed Conversational Applications Communications of the ACM, 43 (9), September 2000 p.53–57
14. http://inova.snv.jussieu.fr/vickie
15. http://www.irit.fr/ACTIVITES/DIAMANT