VoxBoox: A System for Automatic Generation of Interactive Talking Books

Aanchal Jain
The University of Texas at Dallas
Richardson TX 75080
aanchal.jain@student.utdallas.edu

Gopal Gupta
The University of Texas at Dallas
Richardson TX 75080
gupta@utdallas.edu

ABSTRACT

The VoxBoox system makes digital books accessible to visually impaired individuals via audio and voice. It automatically translates a book published in HTML to VoiceXML, and then further enhances this VoiceXML rendering of the book to enable listener-controlled dynamic aural navigation. The VoxBoox system has the following salient features: (i) it leverages existing infrastructure since the book that is to be made accessible need only be published digitally using HTML on the visual Web, (ii) it is based on accepted Web standards of HTML and VoiceXML and thus books can be made accessible inexpensively, and (iii) it is user-centered in that the listener (the user) has complete control over (aural) navigation of the book. In this paper, we present details of the technologies that make the VoxBoox system possible, as well as the details of the system itself. A prototype of the VoxBoox system is operational.

Categories/Subject Descriptors: H.5.2 [User Interfaces];
General Terms: Human Factors

The VoxBoox System

VoxBoox is a system developed by us to enable HTML coded digital books to be made accessible to visually impaired individuals on the fly. The book can be navigated aurally (i.e., using audio and voice) using phones, PDA, mobile and even computers. With the VoxBoox system, blind as well as sighted individuals can easily navigate digitally published books via voice and audio. Because the system can be accessed over the phone, the system can be used to aurally browse books even at times when a computer is not available or cannot be operated (such as while driving).

The VoxBoox system relies on VoiceXML [2] to make digitally published books accessible. VoiceXML is a W3C standard mark-up language for marking up documents that are to be played using audio and that receive input via voice. VoiceXML is the audio/voice analog of HTML, with the voice browser as the aural analog of the visual browser. However, before we discuss how content can be made available in audio/voice via VoiceXML, we need to consider the following issues.

First, there are billions of HTML coded web pages available today on the visual Web. Making all these pages also available in VoiceXML would be expensive and cumbersome, if proper methodology is not used. Indeed, companies have designed voice portals from scratch (such as those by Tell Me Studio) to make certain type of information available in audio/voice (stock quotes, directions, sports scores, etc). Voice portals have not proliferated due to the considerable expense involved in developing aurally navigable pages from scratch. A better (inexpensive) approach, that we have pursued, is to automatically translate HTML coded pages into VoiceXML coded pages [3]. Following this path, the current Web infrastructure can be leveraged to make, at least theoretically, the entire Web content accessible to blind individuals [6].

Second, aural browsing is inherently sequential. This is in contrast to visual browsing where many things can be quickly scanned in one visual glance. Thus, it is of utmost importance to provide extensive navigational control (such as skipping text, moving backwards and forwards) to the user during aural browsing. Along with inherent limitation of sequentially, there are other limitations that are present in VoiceXML itself which significantly curtail users' navigational freedom during aural browsing. VoiceXML is a mark-up language, which has plain text included within "form" tags. A "goto" tag provides navigation from one form to another. Unfortunately, this kind of design leaves the control of how a VoiceXML page is navigated completely in the hands of the writer of the Web page. The user has absolutely no say, apart from providing responses to questions asked during navigation. Given that the author of a VoiceXML page cannot guess all possible ways in which a user might interact with a page, the approach in which the author of the page tries to guess all possible navigation scenarios in advance does not work either. In order to avoid serial reading, which could be boring, as well as giving more flexibility to the listener, adequate navigation controls should be provided to the user. One-way to solve this problem is to enhance the page with additional control facilities during the translation of the document so that the user has a better browsing experience. This is the approach we take [5]. The VoiceXML page is enhanced so that voice commands such as skip (move to next form), back (move control to previous form), start (go to beginning of document), end (go to end of document), repeat (repeat the current form from the beginning) and pause (suspend reading until user says resume) will work during browsing. Additionally, to permit free form navigation, the listener can place speech book-marks (we term them voice anchors) on various forms (paragraphs). Then by uttering these book-marks later, users can move to an arbitrary paragraph in the document at their will. We also allow

ASSETS'06, October 22–25, 2006, Portland, Oregon, USA.
ACM 1-59593-290-9/06/0010.

users to book-mark the whole VoiceXML page, so that they can navigate back and forth between different pages just by uttering these book-marks. Finally, keyword based search is also permitted: users can utter a keyword and navigation will jump to the form that contains that keyword. In all cases, the user may have to spell the word the first time they speak it while placing anchors or book-marks, or during keyword based search. Note that the approach we take satisfies all the criteria under which we wanted to design our system:

- Because contents are obtained by translating HTML pages, existing infrastructure is leveraged;
- Because we obtain content by an automated translation, no effort is required to make a newly published book accessible. As soon as it is available digitally in HTML, it becomes available in VoiceXML as well.
- Because we rely on HTML and VoiceXML, widely accepted standards for the Web, no new language or convention needs to be learned
- Because we enhance the VoiceXML pages with additional control and voice anchors, listeners have complete control over their browsing experience.

Usage Scenario for the VoxBoox System

To understand the VoxBoox system, consider an example scenario. Let us say the user wants to read the book titled “Oliver Twist.” The user will dial a toll free number, which will connect him/her to a Voicebrowser. The first page encountered by the user is an index page, which seeks input from the user. The author will speak the title of the book (Oliver Twist). Because of the limitations of general speech recognition, however, the inputs may have to be spelled the first time. This is very similar to retrieving books over the visual Web, except that the input is via voice and output is via audio. This query is transformed into a search over the regular Web, from where the requested book (published in HTML) is retrieved. The HTML coded text for the book is then automatically translated into the VoiceXML format by our system, which is then played to the user who can listen to the book. While translation, the generated VoiceXML page is automatically augmented with features to allow dynamic navigation of its contents through speech commands and Voice anchors. Speech commands allow the user to move forward and backward, skip and repeat text, etc. They also allow the user to perform a keyword based search. Additionally, users can place speech labels as book-marks (called voice anchors) on passages and return to them later. Using Voice anchors, a listener can mark important passages, and move back and forth between different passages, much like flipping pages of a physical book.

VoxBoox System Architecture

Next we present the architecture of the VoxBoox system (See figure 1). When the user dials a toll free number, he/she is connected to the Voice browser and the menu page is played (the names of the books available is read out one by one; the user can of course aurally navigate this menu page as well). The user utters the name of the book they want to access (there are other options available as well, such as searching by author, etc.); the browser then calls a CGI (common gateway interface) executable to handle the request. A URL is generated on the fly and is passed on to the transcoder (the module that performs the translation of HTML to VoiceXML, developed by us [3]), which fetches the HTML Web page of the corresponding book from the Web server (we use the Web server at online-books.library.upenn.edu). The transcoder then converts the HTML page to VoiceXML and sends it to the enhancer (the enhancer adds control support for skip/pause/etc. as well as voice anchors to the VoiceXML page), which after enhancing sends it back to the transcoder, which sends it back to the CGI and then to the Voice browser . The enhancer, also developed by us [5], inserts code and tags to allow users to place dynamic anchors in various paragraphs and retrieve them later on the fly. It also explicitly adds tags and code to allow users to skip through the paragraphs in the document. Users can repeat sections, go to the end or beginning of the document and can pause the document whenever they want. The user can even go back and forth between pages hyperlinked to each other, just as in the visual Web. The CGI sends the page to the browser and saves it in the history for further reference. The browser then plays the page to the user who can now navigate it aurally.

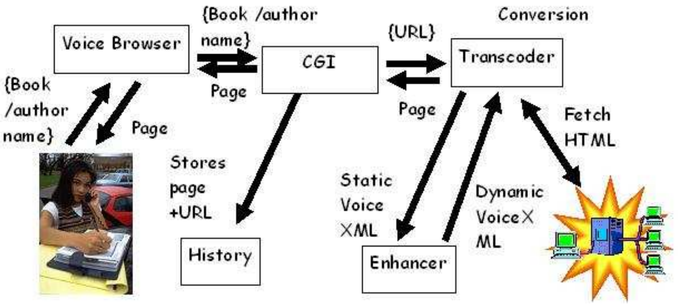

Figure 1: Architecture of Interactive Talking Books

A prototype of the VoxBoox system is operational (visit `http://www.utdallas.edu/~gupta/voxboox/` for a demo). Our approach is better than existing ones, namely, books on audio casettes & CDs (VoxBoox provides voice based controlled and is available over the Internet) and DAISY Digital Talking Books [1] (require publishing in the DTB electronic format and can only be accessed on a computer or using a DTB player, not over the phone). More details of the VoxBoox system can be found elsewhere [4].

1. REFERENCES

[1] The DAISY Consortium. `http://www.daisy.org`.

[2] S. McGlashan et al. Voice Extensible Mark Language 2.0. `http://www.w3.org/TR/VoiceXML20/`.

[3] N. Annamalai, G. Gupta, B. Prabhakaran. An Extensible Translator for translating HTML to VoiceXML. In Proc *ICCHP'04*. pp. 339-346.

[4] A. Jain. Automatic Generation of Interactive Talking Books. Master's thesis. University of Texas at Dallas. Forthcoming.

[5] H. Reddy, N. Annamalai, G. Gupta. Listener-controlled Dynamic Navigation of VoiceXML documents. In Proc *ICCHP'04*. pp. 347-354.

[6] G. Gupta, S. Sunder Raman, M. Nichols. DAWN: Dynamic Aural Web Navigation. In Proc *HCI'05*.