WEBTOUR: A SYSTEM TO RECORD AND PLAYBACK DYNAMIC MULTIMEDIA ANNOTATIONS ON WEB DOCUMENT CONTENT

Chellury R. Sastry
Siemens Corporate Research
755 College Road East
Princeton, NJ 08540
Tel: (609) 734-3317

sastry@scr.siemens.com

Darrin P. Lewis
Siemens Corporate Research
755 College Road East
Princeton, NJ 08540
Tel: (609) 734-6551

dpl@scr.siemens.com

Arturo Pizano
Siemens Corporate Research
755 College Road East
Princeton, NJ 08540
Tel: (609) 734-3614

arturo@scr.siemens.com

ABSTRACT

In this paper, we present the key ideas involved in the design and implementation of "WEBTOUR," a system to record, playback, store, search, and distribute personalized dynamic multimedia annotations on web documents in the form of guided web tours. Using WEBTOUR, web document designers can easily augment the contents of standard web documents by recording such annotations on each document. Upon playback, all media components that comprise the annotation are synchronized and overlaid on top of the web documents. We also present several application scenarios where WEBTOUR is both useful and cost effective.

Keywords

WEBTOUR, guided web tour, multimedia annotations, Plug-In.

1. INTRODUCTION, PRIOR ART, AND MOTIVATION

It is abundantly clear by now that the Internet/Intranet provides a cost effective and ubiquitous medium to connect people and organizations worldwide to exchange and share information via web documents with rich multimedia content. In many situations, just delivering static web content to users may not fully serve their needs. There are a myriad of Internet technologies such as dynamic HTML that can make web pages more interactive, but mere interactive documents may not always serve the needs of users. To understand why this is the case, we consider the following scenarios.

Imagine a user entering the home page of a fairly complex web site, comprised of hundreds of hyperlinked web pages. In such complex web sites, there is a wealth of information that could be beneficial to the user, but in the absence of additional navigational aids, in a casual browsing session, a user may not ever get to the

information he/she needs. The owner of this web site may be a large retail chain presenting information about their merchandise. If users cannot get to some key pages, embedded somewhere deep inside, the benefits of advertising a product on that page may not accrue to the retailer.

There are several navigational aids available, including guided tours of a web site, which can take the customer through key pages (automatically going from one page to the next). For example, using streaming technologies, users can be guided through a web site [1]. A key limitation of such systems is that even though users are guided through the web site automatically, the pages themselves are "static." Overlaying dynamic annotations onto the web pages could further aid the customer in digesting the information presented. The usefulness of making annotations on hypertext documents is a topic of research interest [2, 3]. In commercial systems such as [4], it is possible to overlay annotations in the form of static graphics over web documents.

More and more companies are utilizing the WWW and Internet-based technologies to aid in their customer service, help desk, and call center operations. Web-based call center software solutions enable a company's customers to browse its web site, to make purchases, to post an e-mail query, or to request immediate contact with the right call-center agent. Web-based call center systems, such as the one in [5], in which a customer and call center agent can invoke a synchronized browsing session, could greatly benefit customers. However, in order to cut costs, it is desirable to minimize direct involvement of call center agents with customers, while at the same time providing adequate customer service. To this end, additional tools and aids must be provided to call center agents.

In the realm of distance learning, the use of streaming technologies to download videos of classroom lectures is common. A more interactive and bandwidth-conservative alternative, as discussed in [6], would be for a professor to place the lecture notes on a web server. The professor and students could then exchange knowledge by making annotations on these web pages.

Notice that in the scenarios discussed above, we outlined the important role that adding annotations on web documents can play. However, the kind of annotations considered in [2, 3, 4] are "static," i.e., overlaying graphics or typed text on web pages. The annotations are not rich in multimedia content. We feel that compared to static annotations, users are better served by augmenting the web documents with personalized dynamic

multimedia annotations. In this paper, we present some key ideas involved in the design and implementation of such a system that we refer to as WEBTOUR.

In the next section, we present a detailed overview of WEBTOUR. In Section 3, we present the architecture of WEBTOUR including technical details of the server and client side components, and finally in Sections 4, and 5, we make acknowledgements and present some concluding remarks respectively.

2. WEBTOUR OVERVIEW

Our WEBTOUR system enables recording and playback of dynamic multimedia annotations on a displayed web page. Dynamic multimedia annotations include:

- Mouse gestures and drawings used to call attention to specific areas of a page as a the user browses through it;

- Audio comments and video segments that further describe the page or relate questions about its contents; and

- Hyperlink traversals, which give the author of the annotation the ability to guide the viewer in the navigation of a web site.

A sequence of web pages, having been annotated as such, is referred to as a guided web tour. Upon playback, all of the above user actions are synchronized so that they are rendered exactly as they were recorded. This ability of our WEBTOUR system to synchronize all user actions as they are overlaid on the underlying web documents is what makes WEBTOUR unique (as explained in section 3) when compared with other similar systems.

For example, in Figure 1, we show three pages from a call center web site describing the installation of an audio system.
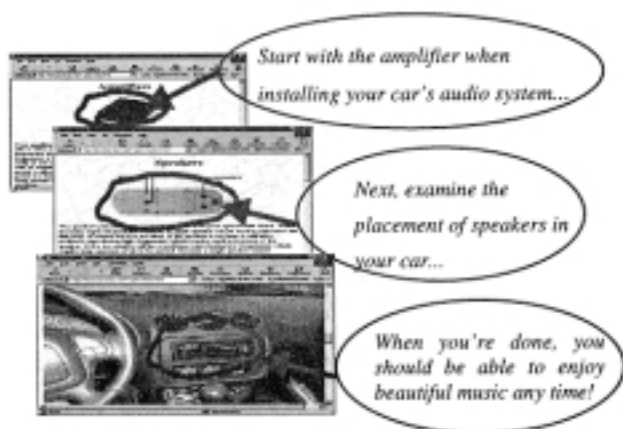

Figure 1. Customer query to a call center answered through a guided web tour

In Figure 1, we see static snapshots of the outcome of dynamic annotation. The user drew on the web page, while at the same time making audio comments shown as italic text in the ovals. All of the captured actions, which include drawing and gesturing with the mouse, speaking the audio commentary, and traversing the hyperlinks from the first page, to the second, to the third, constitute a guided web tour. As mentioned above, these actions are stored while recording and are synchronized and rendered upon playback, resulting in a smoothly flowing presentation.

Since almost all users will be utilizing the ubiquitous Netscape or Internet Explorer (IE) browsers to view web pages, WEBTOUR facilitates the recording and playback of multimedia annotations within these commercial browsers. Since the internal software modules of IE and Netscape are not available to a developer/programmer, providing the ability to record and playback multimedia annotations on these browsers without interfering with their normal functionality poses challenges. This, together with the notion of using dynamic multimedia annotation, is a key contribution of this work.

The key benefits of WEBTOUR include the following:

1. It is a near natural mode of communication in the absence of face-to-face contact between two parties.

2. A guided web tour involves rich multimedia content and the amount of network bandwidth required to download a guided web tour is considerably lower than that required to download other rich multimedia content, such as video.

3. It enables users to work at their own pace.

4. It embodies an easy-to-use authoring paradigm for augmenting and personalizing existing web content.

It may be tempting to conclude that multimedia annotations on web documents can be easily recorded and played back using tools such as Lotus ScreenCam. However, doing so would result in the storage and transportation of large amounts of video data, and at the same time would significantly diminish the interactivity offered by WEBTOUR.

Now that WEBTOUR has been introduced, we highlight how it can be used in some of the scenarios discussed in Section 1. In the call center scenario the direct involvement of call-center agents could be minimized by using WEBTOUR to prepare guided web tours which answer specific queries and other frequently asked questions. Numerous customers can then download and playback the prepared responses.

In the distance learning scenario, professors/lecturers can use WEBTOUR to prepare guided web tours on HTML formatted course materials, as if he/she were teaching on a board. Students can then download these guided web tours at their own paces.

The call center and distance learning application scenarios are by no means the only situations where WEBTOUR can prove expressive and cost effective. We merely highlighted two scenarios where the usefulness of WEBTOUR is almost self-evident.

3. WEBTOUR ARCHITECTURE

As described in the previous section, there are several application scenarios where WEBTOUR can be useful. However, in order to discuss all the features of WEBTOUR, we present its architecture, as it applies to the call center scenario discussed before. This architecture can be easily extended to other applications. The key components of our WEBTOUR system for a call-center are shown in Figure (2). As can be seen from this figure, WEBTOUR is comprised of a set of components on the server side and a set of components on the client side. At the server end, the key components are:

- HTML document registration tool to facilitate the addition of multimedia annotations.

- Annotation databases to store, index, and retrieve guided web tours.

- Indexing software to search through guided web tours.

At the client end, the key component is

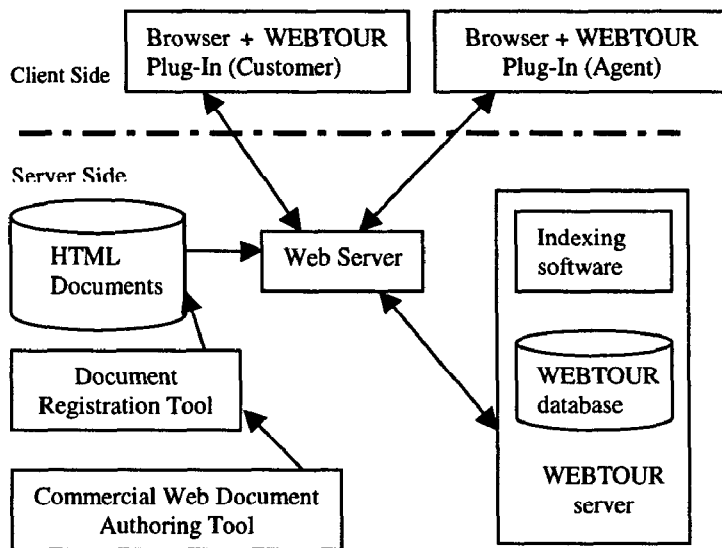- WEBTOUR "plug-in" for recording and playing back annotations on Netscape or Internet Explorer.

Figure 2. Architecture of a call-center WEBTOUR system

We describe each of the above components in some detail in the next two sections.

3.1 Server Side Components

Within a commercial browser, user actions such as mouse clicks are trapped to facilitate navigation. Likewise, such user actions need to be trapped by WEBTOUR in order to facilitate annotation. One elegant technique to trap user action, while at the same time not interfering with the navigation process, is to insert "Javascript" code into each document. This Javascript code traps user action and via the DHTML document-object model, passes that action along to the client-side annotation plug-in (details of this will be described Section 3.2) for recording. The process of adding JavaScript code to web documents before placing them on the web server is referred to as document registration, and it is implemented in a stand-alone component on the server side.

The WEBTOUR database and indexing software together are referred to as the WEBTOUR server. The database stores guided web tours composed by call-center agents. These guided web tours may have been composed as responses to customer queries, or as answers to "Frequently Asked Questions." The WEBTOUR database is implemented using commercial relational database technology, and the Structured Query Language (SQL). SQL statements are used to create an empty skeleton database. Then, after a web tour has been authored, the in-memory data structures of the "plug-in" are normalized into relational format, and inserted into the database using SQL statements. Several relations are needed to represent the complex structure of a web tour. One relation contains information about the mouse movement and drawing events, hyperlink following events, text typing events, and keyword recognition events. Another contains the URLs of web pages. Another contains information about the audio

narration. Still others contain the recognized keywords and words to ignore. "Select" and "join" operations are used to combine and filter the data from these relations. B+tree indexes, provided by the database software, are used to optimize the query processing and shorten search times. The keyword lookup query is a complex query, which involves joins and several indexes.

Given the number of web tours to be stored in the WEBTOUR database, one may wonder how to index the stored web tours such that they can be retrieved and played back. Of course keywords can be chosen from the annotated documents, but more can be done. The web tours contain additional information in the form of recorded speech, drawings, and gestures. While we have made some progress extracting information from all of this content, we have seen tangible results by using speech recognition on the narration. Words can be spotted, in the audio commentary, that serve as keys to index web tours and their associated web pages. We have designed and implemented an activeX control using the IBM Via Voice speech recognition engine and SDK. (Any other speech recognition engine could also be used). At the server end, the audio part of each web tour is extracted as a "wav" file and passed on to the speech recognition activeX control that is embedded in a simple Visual Basic GUI application to extract in text form all the spoken words. Insignificant words are stripped out using a "stopword" list, and the remaining words are associated with the web tour. A search interface is provided to retrieve tours by keyword.

3.2 Client Side WEBTOUR Plug-In
The Client-side "WEBTOUR Plug-In" shown in Figure 2 forms the heart of our WEBTOUR system. Figure 3 shows the Explorer (IE) and Netscape browsers loaded with this WEBTOUR Plug-In and a web page overlaid with multimedia annotations, (The arrow markings indicate mouse drawings and the text in ovals represent audio commentary as the user makes these drawings). We use the generic term WEBTOUR Plug-In even though in the case of IE, this WEBTOUR Plug-In is implemented as an activeX control, while in the case of Netscape, it is implemented as a Netscape plugin. As can be seen from this figure, the WEBTOUR Plug-In has several control buttons. In Figure 3, we show these controls as part of the main browser window in a separate frame, but in general, the WEBTOUR Plug-In along with its controls can be part of another window.

The various functionalities of the Plug-In that result from clicking different control buttons shown in Figure 3 are as follows:

RECORD: All mouse action performed by the user on the browser gets recorded via the WEBTOUR Plug-In. At the same time the user can use the microphone to record his/her audio comments.

PLAYBACK: Any guided web tour in memory is played back. Note: a guided web tour in memory could be one that is composed by the user or one that is downloaded from a web site.

STOP: Clicking this button stops a recording session if user is in recording mode and stops a playback session if user is in playback mode.

PAUSE: Clicking this button puts the user in pause mode. The pause button can be clicked either during a recording session or a playback session.

177

UPLOAD: Clicking this button "uploads" a recorded annotation back to the web server.
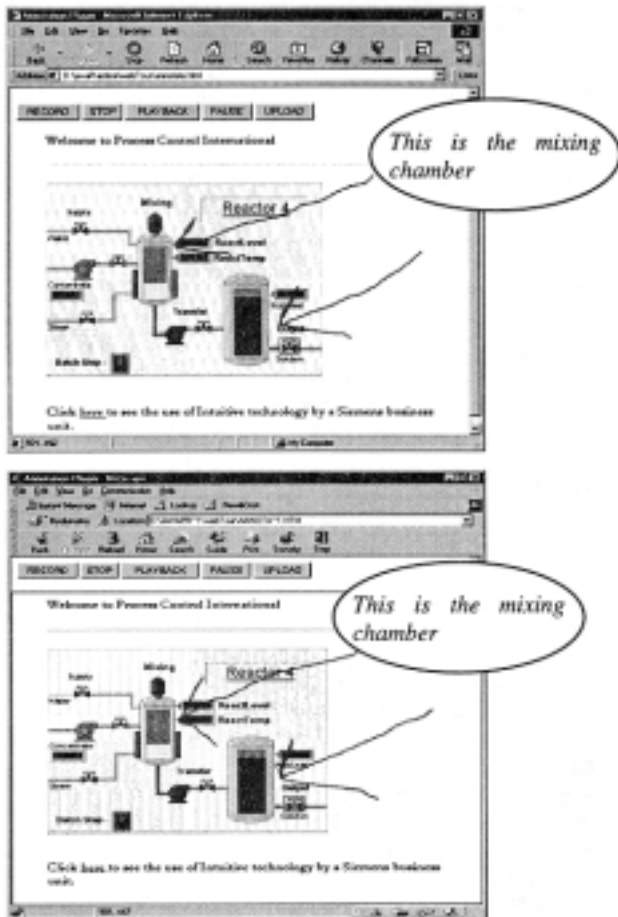

Figure 3. IE and Netscape browsers loaded with WEBTOUR Plug-In and a web page with some multimedia annotations.

A guided web tour can be loaded into a supported browser, just as any other page, given a special URL for that tour. The URL for a web tour, invokes a server-side active server page (ASP) script that delivers a frameset containing (a reference to) the WEBTOUR Plug-In and the initial HTML page in the tour. Next, the ASP script downloads all the data required to playback the web tour into the Plug-In. The user may control the playback of the web tour by using the GUI buttons on the Plug-In.

During playback, the user is free to pause the tour and browse by using traditional browser controls and by following hyperlinks. This is a major advantage of WEBTOUR implementation within the browser.

4. Conclusions and Future Work

In this work, we have presented a system based on the novel idea that users are better served by augmenting web documents with dynamic personalized multimedia annotations rather than just static annotations. The system presented would let a web designer easily record a multimedia annotation and the end user to easily playback these annotations. Furthermore, WEBTOUR is an improvement upon existing systems in that the annotation contents are rich in multimedia and are synchronized upon playback. Finally, the entire WEBTOUR system is implemented using open technologies (standard browsers, Java, Dynamic HTML/JavaScript, Active Server Pages, etc) and hence can be easily customized to suit different applications.

Currently, we are working on several enhancements to the current system. First, as mentioned in Section 2, there are several applications where several users may invoke a synchronized browsing session. The ability for these users to overlay multimedia annotations as they browse would be a significant improvement to such systems.

The use of synchronized multimedia integration language (SMIL) to author multimedia presentations using media streaming is becoming commonplace [4]. We are currently working on a system to use SMIL and WEBTOUR concepts to prepare multimedia rich presentations that can be streamed across the Internet.

5. ACKNOWLEDGMENTS

6. REFERENCES

[1] Synchronized multimedia presentation, including guided web tours. http://www.v-net.com/media.htm.

[2] *Morgan N. Price et. al*, Linking By Inking: Trailblazing in a Paper-like Hypertext, Hyper Text 98, 30-39.

[3] *Catherine C. Marshall*, Toward an ecology of hypertext annotation, Hypertext 98, 40-49.

[4] Annotations in the form of graphics overlaid on web pages. http://www.aspect.com/.

[5] Synchronized browsing between a customer and a call center agent. http://www.wacx.co.uk:8080/product-intro.html.

[6] *Frank M. Shipman III et. al*, Using Paths in the Classroom: Experiences and Adaptations, Hypertext 98, 267-276.

[7] *Robert Stanek*, SMIL: The New Web Format For Multimedia by *By William*, 2/9/99 issue of *PC Magazine*.

[8] *Michael Wynblatt et. al*, A System and Method for Authoring, Distributing and Replaying Derivative Hypermedia Content, Docket No. 98P7974.