Streaming Speech^3: A Framework for Generating and Streaming 3D Text-To-Speech and Audio Presentations to Wireless PDAs as Specified Using Extensions to SMIL

Stuart Goose, Sreedhar Kodlahalli, William Pechter, Rune Hjelsvold
Multimedia Department
Siemens Corporate Research, Inc.
755 College Road East, Princeton, NJ, 08540
Tel: 1-609-734-6500

{sgoose, sreedhar, wpechter, runehj}@scr.siemens.com

ABSTRACT

While monochrome unformatted text and richly colored graphical content are both capable of conveying a message, well designed graphical content has the potential for better engaging the human sensory system. It is our contention that the author of an audio presentation should be afforded the benefit of judiciously exploiting the human aural perceptual ability to deliver content in a more compelling, concise and realistic manner. While contemporary streaming media players and voice browsers share the ability to render content non-textually, neither technology is currently capable of rendering three dimensional media. The contributions described in this paper are proposed 3D audio extensions to SMIL and a server-based framework able to receive a request and, on-demand, process such a SMIL file and dynamically create the multiple simultaneous audio objects, spatialize them in 3D space, multiplex them into a single stereo audio and prepare it for transmission over an audio stream to a mobile device. To the knowledge of the authors, this is the first reported solution for delivering and rendering on a commercially available wireless handheld device a rich 3D audio listening experience as described by a markup language. Naturally, in addition to mobile devices this solution also works with desktop streaming media players.

Categories and Subject Descriptors

H.5.5 [**Sound and Music Computing**]: Modeling; Methodologies and techniques. H.5.2 [U**ser Interfaces**]: Voice I/O.

General Terms

Design, Human Factors, Languages.

Keywords

3D audio, spatialization, speech synthesis, streaming, SMIL, mobile, wireless, PDA, accessibility, location-based.

1. INTRODUCTION

The World Wide Web (WWW) has enjoyed phenomenal growth over recent years and now accounts for a significant proportion of all Internet traffic. The unmitigated success of the WWW bears testimony to the previously unsatisfied need for a system able to integrate and deliver distributed information. WWW support for multimedia was initially limited to text and various image formats. Media files types not intrinsically supported by the browser were downloaded in their entirety and, if available, the appropriate media type application was executed for rendering. However, over the last six years we have witnessed the emergence of streaming media and the subsequent maturation of this technology. Analysts measured that approximately 30 million and 13 million multimedia streams were accessed via the two most popular solutions respectively during August 2001 [19] for content including television/movie trailers, news and radio.

Contemporary streaming media solutions are capable not only of streaming audio and video over low bandwidths, but also to use the native WWW protocol (HTTP) for consumption behind firewalls. Many streaming media players can be embedded within HTML pages and expose interfaces for control and customization via scripting, hence affording a tight integration with WWW browser technology. From late 1996, a W3C activity addressed the specification of a language for choreographing multimedia presentations where audio, video, text and graphics are combined in real-time. This initiative culminated in the Synchronized Multimedia Interaction Language (SMIL) W3C recommendation [37] in 1998. A number of streaming media solutions now exist that offer varying levels of support for the versions of SMIL and associated modules. During the last year a few streaming media solutions have been unveiled for mobile handheld devices [6, 23, 28], with some vendors including optimizations for mitigating transmission errors over wireless networks.

Over recent years members of the W3C Voice Browser working group and the VoiceXML Forum have worked to recommend a standard markup language, called VoiceXML [39, 40], for specifying the spoken dialogs of interactive voice response telephony applications. Such interactive *voice browsers* make extensive use of speech synthesis and recognition technologies to offer an alternative paradigm that enables both mobile and stationary, sighted and visually impaired users to access the *voice web*. Analysts forecast that the mobile phone ownership is to exceed 1 billion during 2002 [26]. Telephone access to on-line content provides (via VoiceXML, SALT [34] or similar technology) the opportunity to embrace an audience without computers. Predictions for the voice portal market by 2005 have been estimated to be $11 billion [20] and the voice commerce market at $1.2 billion [17].

The "cocktail party effect" [1], the human ability to attend selectively to one or more simultaneous conversations in a noisy environment while maintaining a degree of awareness and

WWW 2002, May 7-11, 2002, Honolulu, Hawaii, USA.

appreciation of background auditory stimuli, has been acknowledged for some time [5]. The ears receive the incoming audio stimuli from our three-dimensional world and hence the eyes are instructed in which direction(s) to look. While contemporary streaming media players and voice browsers share the ability to render content non-textually, neither technology is currently capable of rendering three-dimensional media. One analogy may be to contrast monochrome unformatted text with richly colored graphical content. Both media can convey a message, but well designed graphical content has the potential for better engaging the human sensory system. Hence, the author of an audio presentation should be afforded the benefit of judiciously exploiting the human aural perceptual ability to deliver content in a more compelling, concise and realistic manner.

One focus of our research is to promote and improve audio support without sacrificing mobility or accessibility. One challenge for us since has been to deliver this rich audio environment to mobile users. Due to power, processor and economic constraints, contemporary mobile devices cannot yet dynamically generate multiple text-to-speech (TTS) channels in 3D audio space. Hence, we have overcome these limitations by developing a scalable server-side solution for the on-demand creation, spatialization and transmission of the audio stream to a commercial wireless handheld device. This solution has only server side components and does not require any proprietary technology to be installed on the mobile device. The only requirement is that the mobile device has a streaming media player capable of supporting stereo audio. Naturally, in addition to mobile devices this solution also works with desktop streaming media players.

Once the framework was developed for generating on-demand the 3D speech and audio output and streaming it to a wireless PDA, the next challenge was to consider how a 3D audio presentation could be specified. As alluded to earlier, SMIL is a language for specifying the 2D layout and the synchronized time-based presentation of multimedia content. As SMIL possesses many of the constructs required for our task, it was a logical step to leverage SMIL and judiciously extend the language to support:

- in the layout a 3D coordinate system
- in the layout fixed positions in the 3D space
- in the layout the mathematical expression of a trajectory through the 3D space
- in the body a sequence of movements through the 3D space with optional transition styles from one to the next
- in the body an additional medium of text-to-speech

It should be noted that the authors are *not* proposing a general solution rich enough to support all 3D media within SMIL, but currently only the extensions that made sense to us for 3D audio.

This research combines the elements of text-to-speech and audio from VoiceXML, the synchronized presentation of content from SMIL and the transmission and rendering of audio by streaming media solutions. The unique contributions of this research are the 3D audio extensions to SMIL and the generalized framework able to receive a request and, on-demand, process such a SMIL file and dynamically create the multiple simultaneous audio objects, spatialize them in 3D space, multiplex them into a single stereo audio and prepare it for transmission over an audio stream to a mobile device. To the knowledge of the authors, this is the first reported solution for delivering and rendering on a commercially available wireless handheld device a rich 3D audio listening experience as described by a markup language. Hence, this technology can leverage and serve an existing audience of millions of streaming media listeners with accessible content, of a higher fidelity than that offered by VoiceXML, on their wireless and desktop machines alike. The majority of WWW content is textual, and hence offers the potential to build a server-based solution which dynamically generates 3D audio SMIL files which include, or refer to, existing WWW textual content but that are rendered for listeners in a 3D audio manner.

A survey of the related work is discussed in section 2. An overview of audio spatialization is provided in section 3. The judicious extensions to SMIL are presented in section 4. A detailed description of the system architecture is offered in section 5. Section 6 proposes a selection of potential application domains. Section 7 proposes areas for further research, and some concluding remarks are provided in section 8.

2. RELATED WORK

This review selectively traces the progress of interactive voice browsing, 3D audio interfaces, and hence the confluence of these technologies.

Tim Berners-Lee is quoted as saying "*The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect*." [41] But until recent times, to access computer-mediated information, blind people largely relied upon Braille output devices and software known as *screen readers*. A screen-reading program applies various techniques to access the textual content of application software, as it is neither aware nor concerned with the underlying document structure, such as HTML. As a result, the generated TTS output communicates the raw content to the listener but fails to impart any information regarding the structure of the document. However, much of the context is conveyed implicitly through the document structure and layout of the information. A user can apply their understanding of the HTML document structure to aid orientation, navigation, and ultimately, the location of relevant information. Given that structure is a key aid to the comprehension of a visual document, it is of paramount importance to convey this to the user of a voice browser. A number of researchers have since attempted to rectify previous shortcomings. Various early interactive voice browsing solutions for HTML were reported by Petrie [29], Raman [31], Asakawa *et al* [2], Hakkinen *et al* [13] for the desktop, Wynblatt *et al* [43] and Goose *et al* [12] for the vehicle and telephone.

In early 1997 the Web Accessibility Initiative (WAI) [41] was introduced by the W3C to promote this theme through publishing guidelines and developing associated tools. A voice-centric extension to HTML has also been proposed: Aural Cascading Style Sheets (ACSS) [3]. A Voice Browser Working Group [38] was also initiated in 1998 to reach a consensus on the appropriate markup language requirements for speech technology. In 2000, the industry-led VoiceXML Forum specified the VoiceXML language [39], for which a host of commercial solutions now exist. In an effort to standardize the VoiceXML language, the Voice Browser Working Group and the VoiceXML Forum collaborated to produce a recent W3C recommendation [40]. Goose *et al* [10] describe the Vox Portal and a generic solution for the dynamic bi-directional transcoding between HTML and VoiceXML/VoxML.

Ludwig *et al* [18], using specialized hardware and software, augments a graphical windowing system with corresponding audio output positioned in 3D audio space, hence able to convey foreground background status and other related cues. Sawhney *et al* [35] describe a nomadic application for presenting email, voice and reminder messages. This application employs a clock

metaphor and new messages are presented to the user at the position in the 3D audio space corresponding to their time of arrival. The aim of Schmandt *et al* [36] with AudioStreamer is to enhance the effectiveness of simultaneous listening by exploiting the human ability of separating multiple simultaneous audio streams. An interface allows the audio source of greatest importance to the listener to be made more prominent. Kobayashi *et al* [21] describe a system for relating each section of a document to a point on the perimeter of a circle in a spatialized audio interface. This approach allows the human spatial memory to compensate for the weakness of temporal recall. Streaming Speech^3, in common with the systems above, employs spatialized audio technology, however [18, 36, 21] require dedicated audio hardware and the documents to be pre-recorded prior to use. Nomadic Radio [35] requires proprietary wearable computer for rendering a messaging application. Streaming Speech^3 offers no single application but instead provides a generic platform using a software only spatialized audio solution and able to generating the 3D audio content on demand to serve to commercially available wireless handheld devices.

Goose *et al* describe Wire^3 [11], a desktop 3D interactive voice browser. Wire^3 provides a conceptual model of the HTML document structure and maps it to a 3D audio space. Novel features provide information such as: an audio structural survey of the HTML document; accurate positional audio feedback of the source and destination anchors when traversing both inter-and intra-document links; a linguistic progress indicator; the announcement of destination document meta-information as new links are encountered. Roth *et al* enable visually impaired users to gain an appreciation of the HTML document structure by interacting with a touch-screen. In response to the touch-screen input coordinates, AB-Web [33] generates earcons [8, 4] in the 3D audio space that correspond to HTML entities.

Audio Aura [25] by Mynatt *et al* uses VRML to position earcons [8, 4] in the 3D audio space to convey periphery auditory awareness cues that do not require active participation by the user. Infrared sensors register location information and trigger audio cues which are then heard in the user's wireless headphones. A desktop multimodal framework called Speech^3 is reported by Goose *et al* [9] that augments a 3D VRML browser with speech recognition and TTS technology to enable users to issue spoken commands to complex 3D objects in a scene and receive dynamically generated, parameterized spoken feedback. As VRML proximity sensors demarcate speech-enabled 3D objects, when the user enters/leaves the vicinity of a sensor the corresponding speech grammar is loaded/unloaded by the framework to provide a context-sensitive speech-driven interface to VRML scenes.

Audio Aura [25] and Speech^3 [9] both leverage VRML, but Audio Aura does not have speech technology capability. For Streaming Speech^3 the justification for extending SMIL rather than VRML is: SMIL is less complex to author than VRML; SMIL was designed with streaming media in mind; SMIL was designed for presentations; it was easier to extend SMIL than to customize VRML to achieve our goals.

A spatialized audio rendering of the Palm Pilot calendar application was presented and judged by Walker *et al* [42] to be a successful technique for overcoming both the limitations of the small screen and for providing improved recall. However, as the Palm Pilot currently has limited audio support this was a simulation only.

The unique contributions of Streaming Speech^3 are the 3D audio extensions to SMIL and the generalized framework able to receive a request and, on-demand, process such a SMIL file and dynamically create the multiple simultaneous audio objects, spatialize them in 3D space, multiplex them into a single stereo audio and prepare it for transmission over an audio stream to a mobile device. Although much progress has been made in the area of voice browsing and 3D audio interfaces, as evidenced by the literature review, to the knowledge of the authors this is the first reported software-only general solution for delivering and rendering on a commercially available wireless handheld device a rich 3D audio listening experience as described by a markup language.

3. AUDIO SPATIALIZATION

3.1 A Brief Overview

Cognitive psychologists have conducted many experiments to achieve a deeper understanding of the way in which humans interpret through our ears the cacophony of audio signals encountered. There are many cues in the natural environment that facilitate human spatial audio perception. The primary cues are described below:

- **Volume**: the farther away an object is from the listener, the weaker is the sound. This phenomenon is called *roll-off*.
- **Interaural Intensity Difference (IID)**: a sound emanating from the listener's right will sound louder in the right ear than in the left ear.
- **Interaural Time Difference (ITD)**: a sound emitted by a source to the listener's right will arrive at the right ear approximately one millisecond before it arrives at the left ear.
- **Muffling**: the orientation of the ears ensures that sounds emanating from behind the listener are slightly muffled compared with sounds coming from the front. In addition, if a sound is coming from the right, the sound reaching the left ear will be muffled by the mass of the listener's head.
- **Reverberation**: sound reflections from surfaces are known as reverberation. The listener perceives different effects dependent upon the size and shape of the room and the absorptiveness of the surfaces.

Synthetic sound spatialization is the processing of sound in such a way that when it reaches our ears it reproduces the characteristics of a sound located in a 3D-space external to the listener. The effects described above, and many more, can be modeled by a Head-Related Transfer Function (HRTF) [15]. A digitized mono audio stream can be convoluted with an artificial HRTF to create a stereo audio stream that reproduces the timing, frequency and spectral effects of a genuine spatial sound source. For ideal results an HRTF needs to be tailored to an individual, but a generalized HRTF can still produce pleasing results.

3.2 Empirical Results

Summarized here are the results of previous experiments [11] evaluating commercially available 3D audio software toolkits to establish the limitations of the technology and whether it is perceptually feasible to utilize the entire 3D audio space. We began by examining each axis in isolation. For each axis we played different types of sounds at both static and moving positions. The results of these tests concurred with the human cognition literature, and are summarized in table 1.

Although these preliminary experiments yielded results that dampened our initial ambitions, they provide practical guidelines to 3D auditory interface designers. For example, if it is important that the listener accurately identify the position of sound sources then the x-axis should be used, as results offered by Oldfield *et al* [27] indicate that humans can identify to within nine degrees the location of a sound source along the x-axis.

Table 1: Perceptual limitations to using entire 3D audio space.

Axis	Point Sound Source	Moving Sound Source
X	Participants were able to identify accurately the position	Participants were able to identify accurately and track the position
Y	Participants were not able to identify accurately the position	Participants were only able to track the position with a low degree of accuracy
Z	Participants were not able to identify accurately the position	Participants were only able to track the position with a low degree of accuracy

4. RECONCILING 3D SPEECH AND AUDIO REQUIREMENTS WITH SMIL

As expressed previously, SMIL already defines the constructs for specifying a streaming audio presentation, hence we needed only to consider a few additional extensions for spatializing audio elements in three dimensions and adding TTS as a new medium. The extensions to SMIL suggested in this paper represent one proposal, and the authors welcome more elegant and conformant proposals from the SMIL community. Borrowing from voice browser technology and VoiceXML, it was important for our technology to be able to deliver pre-recorded digital audio but also dynamically generated speech in order to offer a high-fidelity platform for leveraging and rendering existing textual content. In addition to being simple enough to hand craft, SMIL can also be dynamically generated in order to personalize presentations [14].

4.1 Composing the SMIL Presentation on the Server

SMIL was conceived to be driven by the player. The SMIL player downloads, or is passed by a browser, the requested SMIL file. Once parsed, the SMIL player is then responsible for scheduling and coordinating the synchronized streaming of the media comprising the presentation. However, as explained previously, this model was not a viable option as, to our knowledge, there are no open source SMIL players suitable for contemporary mobile devices available for us to extend. But even if there were, due to power, processor and economic constraints, contemporary mobile devices cannot yet dynamically generate multiple TTS channels in 3D audio space. Hence, to overcome these limitations we sought to invert the classical SMIL architecture and perform on-demand the parsing, scheduling and the synchronized dynamic generation of the 3D speech and audio presentation on the server and transmitting the resulting stereo audio stream for rendering by the streaming media player on the wireless device.

Naturally, this solution has advantages and disadvantages. The most notable advantage being that it enabled us to succeed in our goal of delivering a 3D audio presentation to a wireless PDA. Another significant advantage is that this solution did not require any Siemens proprietary technology to be installed on the client device. The only general requirement is that the mobile device has a streaming media player capable of supporting stereo audio. In addition to mobile devices, this meant that the solution also works with desktop streaming media players. One disadvantage, however, is that the interactivity provided by a SMIL player is not available as the client player operates simply as a media player.

4.2 Extending the Layout for 3D Audio

Again, a stated aim was to introduce mechanisms for the three dimensional audio space to express trajectories, fixed positions and transitions. Instead of inventing a suite of new elements for this purpose, as much as possible the existing SMIL elements and attributes were reused. However, reaching the current formulation was a challenge involving a number of iterations.

From the SMIL specification the Transition and Animation Modules were identified as containing elements with promise for reusability in this context. The Audio Layout Module defines a relative volume attribute for a region, but this becomes obsolete in a 3D audio context where the relative position and orientation of sound emitters to a listener dictates relative volumes. Although the Transition Module defines transitioning effects out of one media object and into the next, it is not suitable for specifying a transition, or sequence of transitions, from one position in 3Dspace to the next.

The philosophy of SMIL divorces the 2D layout of the media presentation located in the document head from the temporal rendering of the media in the body. To be consistent, we sought to introduce mechanisms which extend the layout into 3D to express positions and trajectories. The `regPoint` element was used as a precedent for a new element `regPoint3d` that contains attributes for the 3D coordinate system. The `regPoint3d` elements indicate positions at which audio sources can be located. Alternatively, the definition of positions through the 3D audio space can be expressed mathematically. The new `trajectory` element offers a general mechanism for specifying a parametric equation for each dimension. The attributes allow the `variables` to be identified in the `expression` and appropriately substituted with the range `values` during computation. The aim is for the definition of reusable trajectories, such as an orbit or an arc from one position to another, that can be employed to animate the audio presentation. The code fragment below illustrates the use of these SMIL extensions in the layout.

```
<trajectory id="trajectory1" interval="1">
  <x expression="sqrt((sqr(r) - sqr(x))"
variables="r x" values="-10..10, -10..10"/>
  <y values="0"/>
  <z values="0"/>
</trajectory>
<layout>
  <regPoint3d id="leg1" x="4" y="0" z="0"/>
  <regPoint3d id="leg2" x="5" y="0" z="0"/>
  <regPoint3d id="leg3"
trajectoryName="trajectory1"/>
</layout>
```

While the Animation Module offers the closest match to our requirements, it is not a perfect fit as it does not address 3D and our need is not for visual animations. Despite these conflicts, the essence of the `animate` element was retained in the new element

`animateLoc`. This element animates the location of a media object `from` one 3D position `to` the next. The original `calcmode` attribute still applies, designating either a discrete jump or a linear interpolation between the `from` and `to` positions. The code fragment below illustrates how the 3D location of an audio object is animated first using fixed 3D coordinates and second using a trajectory.

```
<audio src="file://C|\\Groovy.wav">
  <animateLoc from="leg1" to="leg2"
calcMode="discrete" dur="25s"/>
  <animateLoc from="leg2" to="leg3"
calcMode="linear" dur="15s"/>
</audio>
```

4.3 Text-to-Speech Tag

As alluded to earlier, a stated aim was to introduce a mechanism for having text to be rendering as speech. One option was to overload the SMIL text media object. Instead, a new media object was created called `tts` to allow new attributes to be added without creating confusion. Initially, the only new attribute introduced is *voice* for specifying which speaker, or voice font, in the TTS engine is to be used to speak the referenced, or in-line, text. However, additional properties proposed in ACSS [3] such as speed, pitch and stress may also prove useful here. The code fragment below illustrates the use of these SMIL extensions in the body.

```
<tts src="file://C|\\Headlines.txt"
voice="Mary">
  <animateLoc from="leg1" to="leg2"
calcMode="discrete" dur="25s"/>
  <animateLoc from="leg2" to="leg3"
calcMode="linear" dur="15s"/>
</tts>
```

These proposed extensions address many of the issues, but are by no means complete. Default values for the following assumptions are present in the current implementation:

- the orientation of the sound emmitter currently always faces the listener
- the units of the 3D coordinate space
- 3D audio toolkits have differing conceptual models of sound emmitters and listeners - some using ellipsoids while others use cones. Properties of the ellipsoids and cones can be set and the need for this is clear, but it is unclear how to specify these and remain 3D toolkit agnostic.

5. SYSTEM ARCHITECTURE

5.1 Server Architecture

The request from a media player (in our case the Windows Media Player for Pocket PC [23]) causes an event to be fired by the media server (in our case the Microsoft Windows Media Server [24]) to our component which launches an instance of the Streaming Speech^3 server. A component within the server then parses the requested 3D audio SMIL file pre-computing into a data structure the 3D geometry of each media object throughout the presentation. The scheduling engine then takes control and initializes the 3D audio toolkit (in our case RSX [32]) and the Windows Stream Formatter, before proceeding to generate the body of the presentation.

When a `tts` element is encountered, the scheduler instantiates through SAPI (Microsoft Speech API) the TTS engine and passes the text string to be synthesized. The TTS engine generates a stream of WAV data which the scheduler associates with a streaming emitter of the 3D audio toolkit. When an audio element is encountered, the scheduler associates this WAV data with a static emitter of the 3D audio toolkit. The scheduler is responsible for managing the lifecycle of these emitters over the seq and par constructs. At each time interval throughout the generation of the presentation the scheduler updates the 3D coordinates of each emitter with the pre-computed values to move the audio sources on their prescribed path around the 3D listening space.

The 3D audio toolkit mixes the multiple audio inputs in the 3D listening space, as detailed earlier, to produce a stream of stereo WAV data. This stream is associated with the Windows Stream Formatter which compresses and encodes the data in the specified fashion before forwarding it to the Windows Media Server.

The Streaming Speech3 server can either be used to generate and stream 3D audio presentations on demand, or alternatively the Streaming Speech3 server can be used to generate a 3D audio presentation and save it to a file. Files generated using this latter option can also then be published and streamed by the Windows Media Server.

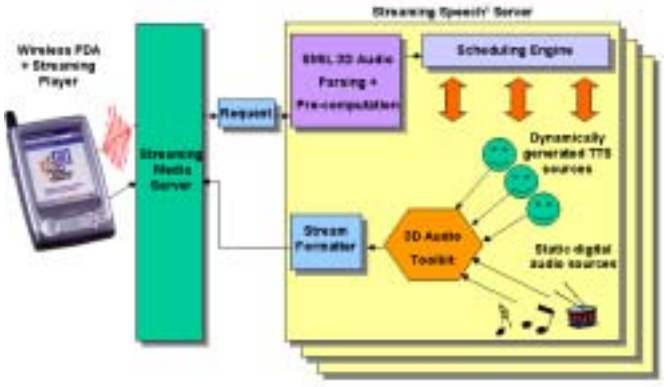

Figure 1: The architecture of the Streaming Speech^3 system.

5.2 Client Requirements

As mentioned earlier, this solution has only server side components and does not require any proprietary technology to be installed on the mobile device. The only requirement is that the mobile device has a streaming media player capable of rendering stereo audio.

The 3D audio is best appreciated through headphones. Due to personal stereos and mobile phones, many people are comfortable with wearing discrete headphones. The latest headphone technology [35, 32] allows stereo audio to be discretely appreciated by the listener but without the need for wearing traditional headphones or earbuds that dominate the ears exclusively.

5.3 Bandwidth Requirements

Tests were conducted to assess subjectively the audio quality over different bandwidths using two wireless technologies prevalent in the USA. The device used was a Compaq iPAQ PocketPC equipped with an expansion jacket. A CDPD (Cellular Digital Packet Data) PCMCIA modem card was tested followed by an 802.11b wireless LAN PCMCIA card. In summary, the

results in table 2 indicate the highest quality audio encoding offered by the underlying streaming platform for the bandwidth of each wireless technology.

With the advent of 2.5G (GPRS) and 3G (UMTS) wireless networks offering increased data rates, applications such as Streaming Speech[3] will soon be able to offer a high quality experience over WANs.

Table 2: Subjective assessment of audio quality.

Techno logy	Maximum Bandwidth	Audio Encoding	Assessment
CDPD	19Kbps	16Kbps, 16KHz, stereo	Providing the signal strength is high enough, the audio quality is acceptable. The speech is intelligible and the 3D effect can be appreciated.
802.11b	11Mbps	96Kbps, 44KHz, stereo	CD quality yielding excellent results

6. POTENTIAL APPLICATION DOMAINS

This technology may be of interest to broadcasters. For example, a talk show host often has multiple panelists. If each panelist is projected appropriately into the 3D listening space (perhaps to model the positions around a table), the listeners are likely to gain an improved conceptual model of the debate. Whereas before the listener had to rely upon the panelists having different sounding voices and a diligent host prefacing questions with the panelist's name, the listener could now associate panelists with positions in the 3D listening space.

Content management, personalization and delivery systems, such as Hot Streams [14], contain much metadata that semantically describes the video content. Using Streaming Speech3, it is possible to augment the video with appropriately positioned additional interjections and sub-commentaries able to enrich the context of the video. Advertisements can also be enriched to create captivating and memorable audio experiences.

If a continual update of position data is sent, using GPS or an indoor tracking technology, the Streaming Speech3 server can adjust accordingly the positions of TTS and audio sources in the 3D audio space to model the user's environment. This has potential in location guidance assistance for drivers and pedestrians. The user could choose to be informed about points of interest, hear which new movies are playing at the nearby theatre, etc.

Options for adding interactivity are described in the following section, and one could then imagine how Streaming Speech3 could be used at the appropriate moment to initiate navigation to an HTML page, VoiceXML or SALT page, or even another 3D audio SMIL file.

7. FUTURE WORK

There are many interesting avenues open for further exploration and below a few of these are elaborated upon. One obvious candidate for further refinement are the SMIL extensions proposed. In particular, omitted from the prototype was the ability to specify the orientation of a sound emitter, the units the 3D coordinate space and to configure the properties of the underlying model, for example the ellipsoids or cones. The ability to specify multiple live audio input sources and position them in 3D space.

Foulke *et al* [8] document the ability of people to listen and absorb speech at fast rates. Results indicate that time compressed speech can be understood by people up to a maximum rate of 275 words per minute. Introducing a words per minute attribute into the `tts` element might prove useful, or perhaps as a parameter specified in the request.

As noted earlier, one casualty in this architecture has been the loss of interactivity that SMIL affords. However, a number of opportunities exist for re-introducing interactivity without requiring any proprietary code on the client side. One solution might be to download an applet to the client browser (exploiting perhaps the Jeode plug-in, or .NET C# code) which maintains synchronization with the server to instruct the browser to render URLs in the browser at the defined times.

8. CONCLUSIONS

It has been posited that by augmenting the underlying streaming platform with 3D text-to-speech and audio capability, content authors, via the proposed SMIL extensions, are now afforded new mechanisms able to better exploit the human aural perceptual ability to deliver content in a more compelling, concise and realistic manner. However, due to power, processor and economic constraints, contemporary mobile devices cannot yet dynamically generate multiple text-to-speech channels in 3D audio space. Hence, the unique contributions of this research are the 3D audio extensions to SMIL and the generalized server-based framework able to receive a request and, on-demand, process such a SMIL file and dynamically create the multiple simultaneous audio objects, spatialize them in 3D space, multiplex them into a single stereo audio and prepare it for transmission over an audio stream to a mobile device. To the knowledge of the authors, this is the first reported solution for delivering and rendering on a commercially available wireless handheld device a rich 3D audio listening experience as described by a markup language. Naturally, in addition to mobile devices this solution also works with desktop streaming media players.

This research combines the elements of text-to-speech and audio from VoiceXML, the synchronized presentation of content from SMIL and the transmission and rendering of audio by streaming media solutions. The convergence of PDA and mobile phone functionality continues to evolve, and technologies such as VoiceXML, SALT and Streaming Speech[3] can compliment one another to serve mobile users. Streaming Speech[3] has the potential to leverage and serve an existing audience of millions of streaming media listeners with accessible content, of a higher fidelity than that offered by VoiceXML, on their wireless and desktop machines alike. The majority of WWW content is textual, and hence offers the potential to build a server-based solution which dynamically generates 3D audio SMIL files which include, or refer to, existing WWW textual content but that are rendered for listeners in a 3D audio manner.

9. REFERENCES

[1] Arons, B., A Review of the Cocktail Party Effect, Journal of the American Voice I/O Society 12, pages 35-50, July 1992.

[2] Asakawa, C. and Itoh, T., User Interface of a Home Page Reader, Proceedings of the ACM Conference on Assistive Technologies (ASSETS), Marina del Rey, USA, 1998.

[3] Aural Cascading Style Sheets (ACSS). W3C Note, http://www.w3.org/Style/css/Speech/NOTE-ACSS

[4] Blattner, M., Sumikawa, D., and Greenberg, R., Earcons and Icons: Their Structure and Common Design Principles, Human-Computer Interaction, 4(1), pages 11-44, 1989.

[5] Bregman, A., Auditory Scene Analysis: The Perception and Organization of Sound. MIT Press, 1990.

[6] Emblaze, http://www.emblaze.com

[7] Foulke, E. and Sticht, T., Review of Research on the Intelligibility and Compression of Accelerated Speech, Psychological Bulletin, 72(1), pages 50-62, 1969.

[8] Gaver, W., Auditory Icons: Using Sound in Computer Interfaces. Human Computer Interaction, 2(2), pages 167-177, 1986.

[9] Goose, S., Gruber, I., Sudarsky, S., Hampel, K., Baxter, B. and Navab, N., 3D Interaction and Visualization in the Industrial Environment, Proceedings of the 9th International Conference on Human Computer Interaction, New Orleans, USA, Volume 1, pages 31-35, August, 2001.

[10] Goose, S., Newman, M., Schmidt, C. and Hue, L., Enhancing Web Accessibility Via the Vox Portal and a Web Hosted Dynamic HTML<->VoxML Converter, Proceedings of the 9th International World Wide Web Conference, Amsterdam, Netherlands, pages 583-592, May, 2000.

[11] Goose, S. and Moller, C., A 3D Audio Only Interactive Web Browser: Using Spatialization to Convey Hypermedia Document Structure, Proceedings of the ACM International Conference on Multimedia, pages 363-371, October, 1999.

[12] Goose, S., Wynblatt, M. and Mollenhauer, H., 1-800-Hypertext: Browsing Hypertext with a Telephone, Proceedings of the ACM International Conference on Hypertext, Pittsburgh, USA, pages 287-288, June 1998.

[13] Hakkinen, M., Issues in Non-Visual Web Browser Design: pwWebSpeak, Proceedings of the 6th International World Wide Web Conference, April 1997.

[14] Hjelsvold, R., Vdaygiri, S. and Leaute, Y., Web-based Personalization and Management of Interactive Video, Proceedings of the 10th International World Wide Web Conference, Hong Kong, May, 2001.

[15] HRTF Measurements of a KEMAR Dummy-Head Microphone, http://sound.media.mit.edu/KEMAR.html

[16] James, F., Presenting HTML Structure in Audio: User Satisfaction with Audio Hypertext, Proceedings of the International Conference on Auditory Display (ICAD), Palo Alto, USA, pages. 97-103, November 1997.

[17] Lernout and Hauspie, http://www.lhs.com

[18] Ludwig, L., Pincever, N. and Cohen, M., Extending the Notion of a Window System to Audio, IEEE Computer, 23(8), pages 66-72, August 1990.

[19] Kelly, T., Internet Media Strategies, http://www.nielsen-netratings.com

[20] Kelsey Group, http://www.kelseygroup.com

[21] Kobayashi, K. and Schmandt, C., Dynamic Soundscape: Mapping Time to Space For Audio Browsing, Proceedings of the ACM International Conference on Computer Human Interaction, Atlanta, USA, March 1997.

[22] Microsoft, (formerly DirectSound) DirectAudio, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnaudio/html/daov.asp

[23] Microsoft, Windows Media Player for Pocket PC, http://www.microsoft.com/windows/windowsmedia/download/pocket.asp

[24] Microsoft, Windows Media Services, http://www.microsoft.com/windows/windowsmedia/en/default.asp

[25] Mynatt, E., Back, M., Want, R. and Frederick, R., Audio Aura: Light-Weight Audio Augmented Reality, Proceedings of the ACM International Conference on User Interface Technology (UIST), Banff, Canada, pages 211-212, October 1997.

[26] Nokia, http://www.nokia.com

[27] Oldfield, S. and Parker, S., Acuity of Sound Localization: A Topography of Auditory Space. I. Normal Hearing Conditions, Perception, 13, pages 581-600, 1984.

[28] Packet Video, http://www.pv.com >

[29] Petrie, H., Morley, S., McNally, P, O'Neill, A and Majoe, D., Initial Design and Evaluation of an Interface to Hypermedia System for Blind Users, Proceedings of the ACM International Conference on Hypertext, Southampton, UK, pages 48-56, April 1996.

[30] Productivity Works Inc, http://www.prodworks.com

[31] Raman, T., The Audible WWW: The World In My Ears, Proceedings of the 6th International World Wide Web Conference, April 1997.

[32] (Formerly Intel) Real Sound Experience (RSX), http://www.radgametools.com

[33] Roth, P., Petrucci, L., Assimacopoulos, A. and Pun, T., AB-Web: Active Audio Browser for Visually Impaired and Blind Users, Proceedings of the International Conference on Auditory Display (ICAD), Glasgow, UK, November 1999.

[34] SALT Forum, http://www.saltforum.org

[35] Sawhney, N. and Schmandt, C., Design of Spatialized Nomadic Environments, Proceedings of the International Conference on Auditory Display (ICAD), Palo Alto, USA, pages 109-113, November 1997.

[36] Schmandt, C. and Mullins, A., AudioStreamer: Exploiting Simultaneity for Listening, Proceedings of the ACM International Conference on Computer Human Interaction, Denver, USA, May 1995.

[37] W3C Recommendation: Synchronized Multimedia Integration Language (SMIL 2.0), http://www.w3.org/AudioVideo

[38] Voice Browser Working Group, http://www.w3.org/Voice

[39] VoiceXML version 1 specification, http://www.voicexml.org/specs/VoiceXML-100.pdf

[40] VoiceXML version 2 specification, http://www.w3.org/TR/2001/WD-voicexml20-20011023

[41] Web Accessibility Initiative, http://www.w3.org/WAI

[42] Walker, A., Brewster, S.A., McGookin, D. and Ng, A., Diary in the sky: A Spatial Audio Display for a Mobile Calendar, Proceedings of IHM-HCI 2001, Lille, France, September 2001.

[43] Wynblatt, M., Benson, D., and Hsu, A., Browsing the World Wide Web in a Non-Visual Environment, Proceedings of the International Conference on Auditory Display (ICAD), Palo Alto, USA, pages 135-138, November 1997.

10. APPENDIX: SAMPLE 3D STREAMING SPEECH SMIL FILE

```
<?xml version="1.0" ?>
<smil3daudio>
<head>
  <trajectory id="trajectory1" interval="1">
    <x  expression="sqrt((sqr(r)  -  sqr(x))"
variables="r x" values="-10..10, -10..10"/>
    <y values="0"/>
    <z values="0"/>
  </trajectory>

  <layout>
    <regPoint3d id="part1" x="4" y="0" z="0"
/>
    <regPoint3d id="part2" x="5" y="0" z="0"
/>
    <regPoint3d id="part3" x="6" y="0" z="0"
/>
    <regPoint3d id="part4" x="7" y="0" z="0"
/>
    <regPoint3d id="part5" x="8" y="0" z="0"
/>

    <regPoint3d   id="leg1"   x="4"   y="0"
z="0"/>
    <regPoint3d   id="leg2"   x="5"   y="0"
z="0"/>
    <regPoint3d                    id="leg3"
trajectoryName="trajectory1"/>

    <regPoint3d  id="staticPosition"  x="0"
y="0" z="0"/>
  </layout>
</head>
<body>
  <seq>
    <audio        src="file://C|\\Theme.wav"
clipBegin="8">
      <animateLoc  from="part1"  to="part2"
calcMode="discrete" dur="5s"/>
      <animateLoc  from="part2"  to="part3"
calcMode="linear" dur="4s"/>
      <animateLoc  from="part3"  to="part4"
calcMode="linear" dur="3s"/>
      <animateLoc  from="part4"  to="part5"
calcMode="linear" dur="2s"/>
</audio>
    <tts      src="file://C|\\Headlines.txt"
voice="Mary">
      <animateLoc   from="leg1"   to="leg2"
calcMode="discrete" dur="25s"/>
      <animateLoc   from="leg2"   to="leg3"
calcMode="linear" dur="15s"/>
    </tts>
    <tts                        voice="Mike"
region="staticPosition" dur="25s">
      Beautiful weather today in Princeton,
but tomorrow...
    </tts>
    <par>
      <audio   src="file://C|\\Ambience.wav"
clipBegin="8">
        <animateLoc  from="part1"  to="part2"
calcMode="discrete" dur="10s"/>
        <animateLoc  from="part2"  to="part3"
calcMode="discrete" dur="20s"/>
        <animateLoc  from="part3"  to="part4"
calcMode="discrete" dur="20s"/>
        <animateLoc  from="part4"  to="part5"
calcMode="linear" dur="10s"/>
      </audio>
      <tts       src="file://C|\\Sports.txt"
voice="Dave">
        <animateLoc   from="leg1"   to="leg2"
calcMode="linear" dur="35s"/>
        <animateLoc   from="leg2"   to="leg3"
calcMode="linear" dur="25s"/>
      </tts>
    </par>
    <audio      src="file://C|\\Closing.wav"
region="staticPosition" dur="22s"/audio>
  </seq>
</body>
</smil3daudio>
```