Audio Engineering Society

Convention Paper

Presented at the 128th Convention
2010 May 22–25 London, UK

Augmented Reality Audio Editing

Jacques Lemordant[1], Yohan Lasorsa[1]

[1] INRIA Rhône-Alpes, France
{jacques.lemordant, yohan.lasorsa}@inria.fr

ABSTRACT

The concept of augmented reality audio (ARA) characterizes techniques where a physically real sound and voice environment is extended with virtual, geolocalized sound objects. We show that the authoring of an ARA scene can be done through an iterative process composed of two stages: in the first one the author has to move in the rendering zone to apprehend the audio spatialization and the chronology of the audio events and in the second one a textual editing of the sequencing of the sound sources and DSP acoustics parameters is done. This authoring process is based on the join use of two XML languages, OpenStreetMap for maps and A2ML for Interactive 3D Audio. A2ML being a format for a cue-oriented interactive audio system, requests for interactive audio services are done through TCDL, a Tag-based Cue Dispatching language. This separation of modeling and audio rendering is similar to what is done for the web of documents with HTML and CSS style sheets.

1. INTRODUCTION

1.1. ARA applications

Augmented Reality Audio (ARA) is used in many mobile applications like geolocalized games, non-linear audio walking tours, navigation systems for visually impaired people. The market is on demand of vertical applications written by a focused developer. Different types of navigation will require different types of applications. A mountain biker navigation application is very different from a guidance application for visually impaired people. The rendering of an ARA scene can be experimented through the use of bone conduction headsets, headphones with integrated microphones or earphones with acoustically transparent earpieces, with the audio being played by a mobile phone. We show in this paper than an ARA scene can be authored through the join use of two XML languages, one for the representation of the real world and the other one for the representation of the 3D virtual audio scene, the glue between the two being done through a tag-based dispatching language.

1.2. Format for mobile interactive 3D audio

We have designed an XML language (A2ML) for interactive audio on mobiles, to follow the trend in powerful mobile audio devices lead by Apple with its iPhone. A2ML offers, concerning the sequencing of sounds, a level of capabilities similar to that of iXMF, the interactive audio file format defined by the

Interactive Audio Special Interest Group (IASIG) [2]. But as opposed to iXMF, A2ML supports not only sequencing but also 3D sound rendering and declarative parameter's animation (DSP and positional parameters). A sound manager for the iPhone has been implemented. This sound manager will be adapted to OpenSymbian and android phones when OpenSL ES [3] will be available. The construction of an A2ML sound model for an interactive jungle, usable in a game or in a sound installation, is explained in [1] and downloadable as a rendered audio file in MP3 format from *http://gforge.inria.fr/projects/iaudio*.

1.3. Two stages iterative editing

Interactive audio authoring systems for games are used in a static manner with the help of auditioning tools, in front of a console because the world where audio sources are placed is a virtual one. These systems are not well-suited for the authoring of an ARA scene for which mobile (non-static) mixing is necessary to apprehend the audio spatialization and the chronology of the audio events which depend on the position of the listener.

Our proposal is to do ARA editing in two stages:

- In the first stage, the localization and geometric characteristics of the sound sources are controlled directly on the mobile using a see-through touch-screen speech interface concept. Communication through speech is a very natural interaction with auditory displays. This verbal command, restricted to a predefined catalogue of words, is the fastest way to interact with sound sources in the scene, modify sound sources locations or DSP parameters.

- In the second stage, a textual A2ML authoring of the sequencing of the sound sources and DSP acoustics parameters is done either directly on the mobile or on a computer. Finally, we alternate between these two stages until a satisfactory result is obtained.

2. FORMATS FOR ARA SCENES

It's tempting to embed sound objects in the format describing the real world but this prevents the user to easily modify the soundscape. For example, in guidance applications for visually impaired people, it's important to let people choose the density of sound objects and the nature of these sound objects, vocal or audio. We will present first the format for geographic data, then the format for interactive audio and then how to reference cues that need to be activated by means of a dispatching language. We will use the guidance application for visually impaired people we are developing at INRIA (Grenoble, France) under the Autonomy project as an example to illustrate the way formats for geographic data and interactive audio can be used to build Augmented Reality Audio applications.

2.1. XML Format for geographic data

A format for a cue-oriented interactive audio system, like A2ML allows using standard geographic formats like OpenStreetMap [4] without embedding sound sources. We choose OpenStreetMap as a format for geographic data because it has a very simple schema and therefore can be easily used to build a node-edge representation of buildings, allowing both outdoor and indoor ARA applications.

The OpenStreetMap modeling of a corridor of the INRIA building in Grenoble is shown in figure 2 together with a graphical view in figure 7.

Figure 1 Aerial view of the INRIA Building in Grenoble

```
<?xml version="1.0" encoding="UTF-8"?>
<osm version="0.6">
    <node id="B11" x="4.00"  lon="5.00"/>
    <node id="B12" x="8.00"  lon="5.00"/>
    <node id="B13" x="16.00" lon="5.00"/>
    <node id="B14" x="22.00" lon="5.00"/>
    <node id="B15" x="28.00" lon="5.00"/>
    <node id="B16" x="36.00" lon="5.00"/>
    <node id="B17" x="38.00" lon="5.00">
        <tag k="barrier" v="firewooddoor"/>
    </node>
    <node id="B18" x="46.00" lon="5.00">
        <tag k="tactile_paving" v="yes"/>
    </node>
    <node id="B19" x="46.00" lon="3.00">
        <tag k="floor_change" v="stairs"/>
    </node>
    <node id="B20" x="46.00" lon="2.00">
        <tag k="floor_access" v="elevator"/>
```

```
        <tag k="elevator" v="button"/>
    </node>
    <node id="S2" x="42.00" lon="3.00">
        <tag k="floor_access" v="stairs"/>
        <tag k="ele" v="14"/>
    </node>
    <node id="C17" x="45.2143789" y="7.00">
        <tag k="anemity" v="toilets"/>
        <tag k="addr:roomnumber" v="B207"/>
    </node>
    <relation id="R1">
        <member type='node' ref="B12"/>
        <member type='way' ref="B216" role="door"/>
        <member type='way' ref="B218" role="door"/>
        <member type='way' ref="B219" role="door"/>
        <member type='way' ref="B217" role="door"/>
        <tag k="type" v="junction"/>
    </relation>
    <way id="B1">
        <nd ref="B11"/>
        <nd ref="B12"/>
        <nd ref="B13"/>
        <nd ref="B14"/>
        <nd ref="B15"/>
        <nd ref="B16"/>
        <nd ref="B18"/>
        <tag k="name" v="Aisle B"/>
        <tag k="floortype" v="carpet"/>
    </way>
    <way id="BE">
        <nd ref="B14"/>
        <nd ref="BE1"/>
        <nd ref="BE2"/>
        <tag k="name" v="gateway B"/>
        <tag k="floortype" v="carpet"/>
    </way>
    <way id="B2"  timestamp="2009-01-06T22:45:27Z">
        <nd ref="B18"/>
        <nd ref="B19"/>
        <nd ref="B20"/>
        <tag k="name" v="Aisle A"/>
        <tag k="floortype" v="marble"/>
    </way>
</osm>
```

Figure 2 OpenStreetMap model of the West second floor of the INRIA Building

2.2. Format for interactive audio

ARA applications use environmental audio, informative speech and sound objects [5] to create a soundscape. A sound object is a time structure of audio chunks whose duration is on the time scale of 100 ms to several seconds. These sound objects have heterogeneous and time-varying properties.

A2ML [1], an XML language inspired by iXMF [2] is well adapted to the creation of soundscape. A2ML can be used as an authoring time interchange file format for interactive mobile applications or as a runtime file format that is actually loaded through the web and played directly in the mobile. A2ML is a cue-oriented interactive audio system where instance of cues (sound objects) can be created using named events and then modified in real time through an API.

The most basic audio element of our hierarchy is the *sound*: a simple container for an audio file. Then we have the *chunk*, an audio fragment made from one or many sounds, or only a part of it. Each time a chunk is requested to play, if it contains multiple sounds, one is picked based on the rules defined. Sounds can be picked in order or randomly, excluding last played sound or not, and the chance of picking a particular sound over the others can be adjusted. On top of that, the *cue* is an identified container for a sequence of one or more chunks, which can be directly managed by the audio artist to be started and stopped when needed.

Cues are the most important elements to the audio artist as they represent the directly playable soundtrack elements. They are models for sound objects, which can be instantiated through external or internal events. The 3-level audio hierarchy of cues was designed to be simple, effective and adaptable to most situations, while leaving space for creativity.

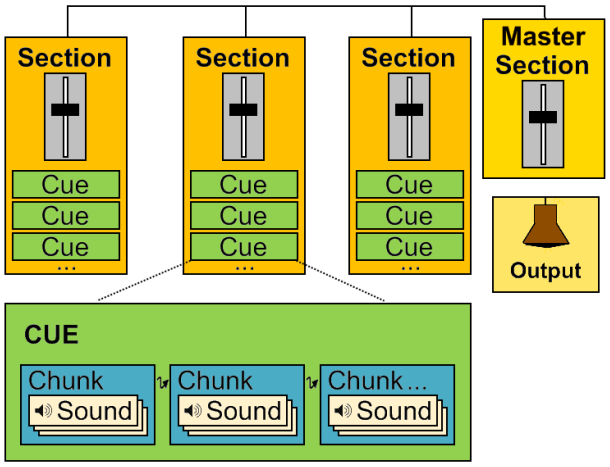

Figure 3 Sound objects in A2ML

Like in a traditional mixing console, mix groups can be used to regroup multiple cues and apply mix parameters on all of them at the same time. In our format, we called them *sections* as, in addition to mixing multiple cues, they can also be used to add DSP effects and locate the audio in a virtual 3D environment. The main difference with traditional mix groups is that a cue can be a member of multiple sections, and the effects of all of them will apply, making sections very versatile.

The sound manager's response to a given cue instantiation may be simple, such as playing or halting a

3D sound source, or it may be complex, such as dynamically manipulating various DSP parameters over time. The sound manager is offering an API through which all cue instance parameters can be manipulated such as positions of the sound sources and the auditor.

In ARA mobile applications, cue events are generated depending on the position of the user. Localization of the user can be computed by a technique called map-aided positioning by using the Micro-Electro-Mechanical Systems (MEMS) embedded in the mobile (accelerometers and magnetometers) or in large space like a hall through communication of the mobile with external wireless sensors. For example, MEMS found in most smart phones allow to compute the number of steps between two way points in a guidance application by monitoring vertical acceleration as shown in figure 4. With the orientation given by the embedded 3D compass, position on the map can be computed. Tactile paving at waypoints can then be use to correct the accumulation of errors.

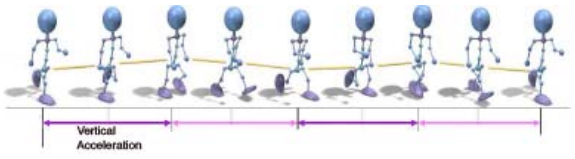

Figure 4 Localization using MEMS embedded in mobiles by computing vertical acceleration

A2ML is interactive an audio format well suited to ARA guidance applications with specific types of navigation. As an example, the A2ML description of a soundscape for guidance of visually impaired people in a part of the INRIA Building is shown in figure 5.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- A2ML testing model for indoor guidance system. -->
<a2ml xmlns="http://wam.inrialpes.fr/a2ml">
    <cues>
        <!-- Next waypoint in the current path. Duration = min(1/dist*factor, 10s). -->
        <cue id="next_wp" loopCount="-1" begin="audioguide.start_guide" end="audioguide.end_guide" dur="10s">
            <chunk>
                <sound src="/guide/next_wp.wav"/>
            </chunk>
        </cue>
        <!-- Vocal indication of the distance to the next waypoint, in steps. -->
        <cue id="distance_to_wp" loopCount="1" begin="audioguide.get_distance_to_wp">
            <chunk pick="fixed">
                <sound src="/guide/distance_1_step.wav" setActive="audioguide.set_distance_to_wp.1_step"/>
                <sound src="/guide/distance_2_step.wav" setActive="audioguide.set_distance_to_wp.2_step"/>
                <sound src="/guide/distance_3_step.wav" setActive="audioguide.set_distance_to_wp.3_step"/>
                <sound src="/guide/distance_4_step.wav" setActive="audioguide.set_distance_to_wp.4_step"/>
                <sound src="/guide/distance_5_step.wav" setActive="audioguide.set_distance_to_wp.5_step"/>
                <sound src="/guide/distance_6_step.wav" setActive="audioguide.set_distance_to_wp.6_step"/>
                <sound src="/guide/distance_7_step.wav" setActive="audioguide.set_distance_to_wp.7_step"/>
                <sound src="/guide/distance_8_step.wav" setActive="audioguide.set_distance_to_wp.8_step"/>
            </chunk>
        </cue>

        <!-- Objects that are part of the guidance path. -->
        <cue id="door" loopCount="1" begin="object.door">
            <chunk>
                <sound src="/object/door.wav"/>
            </chunk>
        </cue>
        <cue id="stairway" loopCount="1" begin="object.stairway">
            <chunk>
                <sound src="/object/stairway.wav"/>
            </chunk>
        </cue>
        <cue id="elevator" loopCount="1" begin="object.elevator">
            <chunk>
                <sound src="/object/elevator.wav"/>
            </chunk>
        </cue>
        <cue id="elevator_button" loopCount="1" begin="object.elevator_button">
            <chunk>
                <sound src="/object/elevator_button.wav"/>
            </chunk>
        </cue>

        <!-- Environment as part of the guidance path. -->
```

```
        <cue id="ambiance" loopCount="-1" begin="environment.ambiance">
            <chunk pick="fixed">
                <sound src="/environment/ambiance_office.wav" setActive="environment.set_ambiance.office"/>
                <sound src="/environment/ambiance_hall.wav" setActive="environment.set_ambiance.hall"/>
            </chunk>
        </cue>
        <cue id="floor_surface" loopCount="1" begin="environment.floor_surface_change">
            <chunk pick="fixed">
                <sound src="/environment/floor_surface_carpet.wav"
                       setActive="environment.set_floor_surface.carpet"/>
                <sound src="/environment/floor_surface_marble.wav"
                       setActive="environment.set_floor_surface.marble"/>
            </chunk>
        </cue>
        <cue id="way" loopCount="1" begin="environment.way">
            <chunk>
                <sound src="/environment/way.wav"/>
            </chunk>
        </cue>

        <!-- Details on the immediate environment that can be requested. -->
        <cue id="atrium_door_number" loopCount="1" begin="details.atrium_door_number">
            <chunk id="door1" pick="fixed">
                <sound src="/details/door_number_B214.wav" setActive="details.set_atrium.atrium1"/>
                <sound src="/details/door_number_B208.wav" setActive="details.set_atrium.atrium2"/>
                <sound src="/details/door_number_B204.wav" setActive="details.set_atrium.atrium3"/>
            </chunk>
            <chunk id="door2" pick="fixed">
                <sound src="/details/door_number_B212.wav" setActive="details.set_atrium.atrium1"/>
                <sound src="/details/door_number_B206.wav" setActive="details.set_atrium.atrium2"/>
                <sound src="/details/door_number_B202.wav" setActive="details.set_atrium.atrium3"/>
            </chunk>
            <chunk id="door3" pick="fixed">
                <sound src="/details/door_number_B213.wav" setActive="details.set_atrium.atrium1"/>
                <sound src="/details/door_number_B207.wav" setActive="details.set_atrium.atrium2"/>
                <sound src="/details/door_number_B203.wav" setActive="details.set_atrium.atrium3"/>
            </chunk>
            <chunk id="door4" pick="fixed">
                <sound src="/details/door_number_B215.wav" setActive="details.set_atrium.atrium1"/>
                <sound src="/details/door_number_B209.wav" setActive="details.set_atrium.atrium2"/>
                <sound src="/details/door_number_B205.wav" setActive="details.set_atrium.atrium3"/>
            </chunk>
        </cue>
    </cues>

    <sections>
        <!-- Mix group for the global audio guide. Use the reverb as a way to notify room size changes. -->
        <section id="audioguide" cues="next_wp door stairway elevator elevator_button ambiance floor_surface
        way">
            <dspControl dspName="reverb">
                <parameter name="preset" value="default"/>
                <animate id="preset_change" attribute="preset" values="environment.change_reverb_preset"/>
            </dspControl>
            <volumeControl level="70"/>
        </section>

        <!-- Activates 3D positioning for the object that need it. Position of the objects is controlled by the
        guidance application. -->
        <section id="objects3D" cues="next_wp door stairway elevator floor_surface">
            <mix3D>
                <distanceAttenuationControl attenuation="2 25 false 1" />
            </mix3D>
        </section>

        <!-- Submix group for the environment details. -->
        <section id="details" cues="atrium_door_number">
            <mix3D>
                <distanceAttenuationControl attenuation="5 50 false 1" />
            </mix3D>
            <volumeControl level="100"/>
        </section>
    </sections>
</a2ml>
```

Figure 5 A2ML soundscape for the auditory display of the INRIA building corridor

2.3. Tag-based Cue Dispatching Language

For an ARA application, we have to find a way to transfer localization events in the real world into cue events. We don't want cues or sound objects to be positioned on a map but activated indirectly through map events. This is done through TCDL, a Tag-based Cue Dispatching Language, reminiscent from NVDL (Namespace-based Validation Dispatching Language) [6]. TCDL allows us to use A2ML as an audio style sheet for OpenStreetMap.

We will use our guidance application to explain concepts behind TCDL. We have to raise cue events depending of the OpenStreetmap feature tags contained in nodes, ways and relations in the proximity of the user. We detect the tags raising cue event by using synthetic radar in front of the user as shown in figure 6.

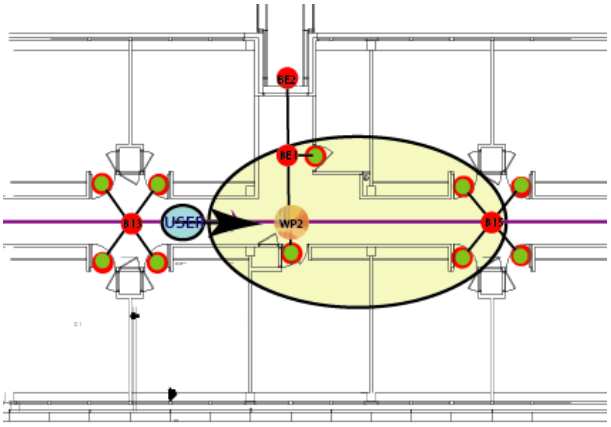

Figure 6 Radar for OSM tags detection

The tags found in the radar zone (in yellow on figure 6) are use to build an audio spatial display including environmental cues, sound objects and vocal announcement. In addition, spatial guidance signals (beeping sounds) are used between the waypoints, which are determined through a shortest path algorithm (Dijkstra's algorithm). In figure 6, the user is hearing a beeping signal coming from WP2 together with the cues, which will be instantiated as specified in the tag-cue mapping described by the TCDL document.

By modifying the TCDL and A2ML documents, we can customize the auditory display, as we like. It is even possible to do audio rendering of more features by adding a special style file (e.g. toilets, bar) or by grouping them in a layer element as shown in figure 7.

```
<?xml version="1.0" encoding="UTF-8"?>
<rules name="standard">
    <!-- Standard Audio Style Sheet -->
    <rule e="node" k="anemity" v="*">
        <layer name="anemity" audible="no">
            <rule k="anemity" v="toilets">
                <cue name="details.toilets" />
            </rule>
            <rule k="anemity" v="bar">
                <cue name="details.bar"/>
            </rule>
        </layer>
    </rule>
    <rule e="way" k="floortype" v="carpet">
       <cue name=" environment.surface.carpet"/>
    </rule>
    <rule e="way" k="floortype" v="marble">
       <cue name=" environment.surface.marble"/>
    </rule>
    <rule e="relation" k="type" v="junction">
       <cue name=" details.atrium_door_number"/>
    </rule>
    <rule e="node|way" k="tactile_paving" v="yes">
        <cue name ="guide_tactile_paving"/>
    </rule>
    <rule e="node" k="floor_access" v="stairs">
        <cue name ="object.stairs"/>
    </rule>
    <rule e="node" k="floor_access" v="elevator">
        <cue name ="object.elevator"/>
    </rule>
    <rule e="node" k="floor_access" v="elevator">
        <cue name ="object.elevator"/>
    </rule>
    <rule e="node" k="elevator" v="button">
        <cue name ="elevator.button"/>
    </rule>
</rules>
```

Figure 7 Tag-based Cue Dispatching Language (TCDL)

3. REAL WORLD VIRTUAL SONIFICATION

3.1. Auditory display for guidance applications

Sound design matters and a language like A2ML whose cues are sound models designed by an audio engineer and instantiated by a programmer is very helpful in this respect.

Our goal in our guidance application is to construct a sound spatial display that provides the user with direct perceptual information about the spatial layout of the environment including the destination.

Impact sounds can be used as audio chunks to build sound objects. Impact sounds on different kind of material (wood, metal, glass) can be classified as coming from impulsive contact, like hitting, breaking, bouncing or continuous contact like rolling and sliding.

Three types of sound sources are needed beacons, environmental sources and sound objects.

Beacons are used for directional guidance, where the user walks directly toward the sound. A 3D rapid beeping sound can be used so that the beeps appear to come from the direction of the next waypoint. The remaining distance to the next waypoint is announced by making a variation of the time between the beeps, depending of the distance (being closer meaning faster repetitions of the beeps). Directional cues are used less in a structured space, as when walking in a corridor, than in other more open situation, where fewer cues are available to guide travel.

Environmental audio sources giving information on the topology of the immediate environment and in a safe environment, environmental audio sources like very light music are used. All sounds, except the vocal guide, can be changed in pitch to reflect the reaching of a potential dangerous place.

By listening to the sound objects, the user gains an idea of his surroundings [7]. Sound objects indicating nearby items of possible interests such as obstacles, portals and useful objects.

- Obstacles are barriers that interfere with the navigation. Sounds with an increased volume when the barrier gets closer can be employed to sense these obstructions

- Portals can be doors, escalators or stairs. Auditory texture provides the information about the ground floor material. Transitional sounds can be employed to announce the transit through a portal. An example of a transitional sound is the sound of a hinge opening played when the user passes by a door. For stairs and elevators, it is helpful to verbally notify when a new level is reached.

- Interactions with objects: buttons power switch, telephone. Functionality has to be explained, either verbally or through describing sounds. Feedback information about the success of a current operation is done by changing the describing sound of the object. Focusing on a specific object is done by suppressing all environmental sounds, and by enabling additional audio cues like oral description.

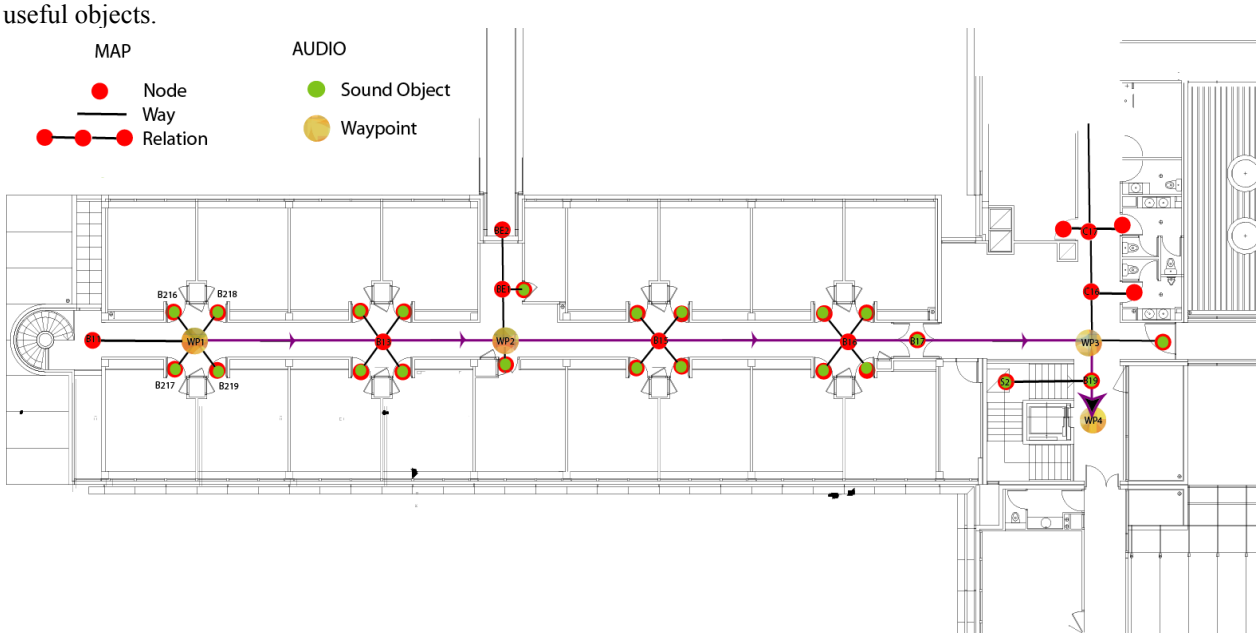

Figure 8 Virtual Sonification of the INRIA Building

4. AUTHORING OF AN ARA GUIDANCE APPLICATION

4.1. Mobile mixing of cues instances

The chronology of events depends on the path of the listener. Thus, the creator of audio scenes must necessarily experience these movements in order to assess his own work and possibly adjust some parameters.

Figure 9 shows an augmented reality view of the soundscape through the camera of a mobile. The user can selects an audio layer if the soundscape is too complex. Small radar with a white viewing angle as in [8] shows all the sound sources in the current range. In the main view, the sound sources are shown in such a way to indicate their positions and respective distances. Concentric rings are used to indicate distance, in the small radar. A very natural interaction is communication through speech and touch. This verbal information exchange is the fastest way to interact with sound sources in a scene, or other DSP parameters. As the sound sources are objects from within the environment, the communication is positional and the speech originates from the sound source location in the scene.

If the auditory display is cluttered with sound objects, a sequential display of the sound objects can be activated. The application automatically selects the closest sound source (SS) and jump from one SS to the next every few seconds. The user can also lock on a SS by selecting it. Information about the currently selected SS is shown. This information consists of a cue instance name and DSP parameters.

Creative work in the context of audio augmented reality is a task requiring a significant period of listening and displacement from the author. This task can be greatly simplified by producing mobile tools that will allow the author to make audio adjustments in situ together with a textual editing of the structure of the sound objects described in the audio format.

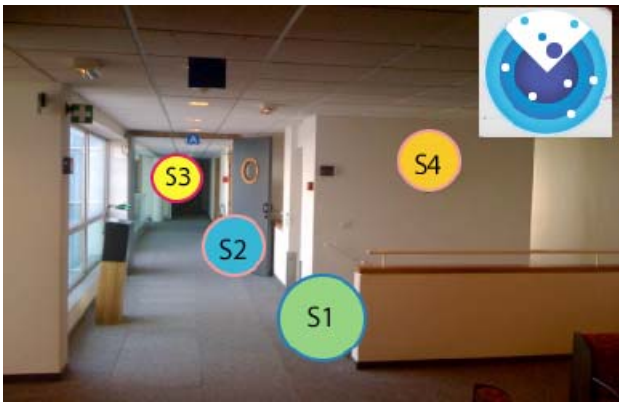

Figure 9 See-through Touch-screen Speech Interface (INRIA Building)

4.2. TCDL and A2ML Editing

All the tools existing for XML-based languages facilitate textual editing of TCDL and A2ML documents. Sound objects models can be designed using an XML editor for people acquainted with markup languages. As it's not the case of most audio designers and at the opposite programmers are not familiar with sound modeling, we are thinking to build a specific editor using templates to guide the design of sound objects with predefined models and consequently ease the process of editing the A2ML language.

5. CONCLUSION

We have shown that the use of XML formats for real world modeling and its sonification is very helpful in the authoring of augmented reality audio (ARA) applications. Moreover, by keeping the geographic and interactive audio formats independent from each other, the application is easily adaptable to users with different capabilities as it's the case in guidance application of visually impaired people.

6. ACKNOWLEDGEMENTS

This work was done under the auspice of the Autonomy Project (Global Competitive Cluster Minalogic) and financed with the help of the European Fund of Regional Development (EFRD).

7. REFERENCES

[1] An Interactive Audio System for Mobile, Yohan Lasorsa, Jacques Lemordant, *127th AES Convention.*

[2] Interactive XMF, Interactive Audio Special Interest Group, http://www.iasig.org./wg/ixwg/

[3] OpenSL ES, http://www.khronos.org/opensles/

[4] OpenStreetMap, http://www.openstreetmap.org.

[5] Sound Objects, based on Pierre Schaeffer's work, http://en.wikipedia.org/wiki/Pierre_Schaeffer

[6] NVDL (Namespace-based Validation Dispatching Language), http://www.nvdl.org

[7] Interacting with sounds, N.Rober and M.Masuch, ICAD 04, 2004

[8] Open augmented reality browser, http://layar.com

[9] Evaluation of Spatial Displays for Navigation without Sight, J. Marston, J. Loomis and all, ACM Transactions on Applied Perception, V3, N2, 2006