Abstract: A system and method are disclosed for providing improved 3D sound experience to a user. The sound generation layer is customizable to allow the user and/or application provider to modify the internal rules the sound generation layer uses to render sounds, to amplify sounds that fall below a pre-set or user-set volume level, and to specifically amplify/soften certain sounds (such as game specific sounds like gunfire or footsteps), or specific frequencies of sounds. A graphical user interface can communicates with the sound generation layer to handle any or all the above, so that a lay user can easily adjust these settings without having to understand the underlying algorithms.
ENHANCED 3D SOUND

FIELD OF THE INVENTION

The present invention relates generally to the field of audio equipment, and more particularly to enhancing the audio experience of users of software applications having 3D sound.

BACKGROUND OF THE INVENTION

Currently, software applications that make extensive use of a 3D sound stage, such as computer games, typically use one of a few general application program interfaces ("APIs"), such as DirectSound3D® from Microsoft Corporation, or the open standard OpenAL, to communicate data. That data includes, for example, the sounds' position relative to the listener, including the direction, height, distance, initial volume, any distortion effects, and the actual audible noise itself. That data is transmitted to the sound software/hardware that is responsible for the actual physical generation of the sound (the "Sound Generation Layer," or "SGL"). The SGL comprises the interface API (e.g., DirectSound3D®), the sound card/hardware's software driver, and the sound generation hardware.

The SGL then handles the actual generation of the physical sound based on the data received from the software application. Based on its internal rules, the SGL may decide to reduce the volume of a sound (e.g., a gunshot) to simulate the sound originating as further away from the listener's position. Alternatively, the SGL may be instructed to play the same gunshot sound but simulated at three different positions, and hence would choose which speakers (in potentially varying configurations), the volume, and the balance at which to play the three gunshots. Depending on how successful the SGL is at this, the listener should believe that these would be three distinct shots fired from three different positions despite being the same identical sound file being played.

Certain computer games, such as first person shooter games, provide a large range of audio cues to inform the player of the game such information like enemy footsteps approaching from the rear, rockets whooshing overhead, or an enemy hiding behind a box but quietly reloading his weapon. All this information is particularly critical in a competitive environment, but users are unable to customize the way this
information is presented to them. Of course, such information makes generally improves the audio experience of the user of this or other audio applications.

It is thus desirable that the user have available a system and method that allows customization. While peripherally, users may adjust their volume controls so that softer, further away sounds may be heard more clearly, or adjust software/hardware equalizer controls to amplify sounds at specific frequencies, neither of these is particularly satisfactory. Adjusting volumes of soft sounds also mean loud sounds are adjusted.

SUMMARY OF THE INVENTION

An improved 3D sound experience may be provided with several components described herein. These are primarily intended to be implemented via changes to the SGL at the sound driver level, although it is also possible to implement at the sound hardware level:

1. Allowing the user to modify the internal rules the SGL uses to render sounds (e.g., how much softer to play sounds that are further away).

2. Allowing users to amplify sounds that fall below a pre-set or user-set volume level, while leaving sounds that are above this level untouched

3. Allowing users to specifically amplify/soften certain sounds (such as, in the case of games, game-specific sounds like gunfire or footsteps), or specific frequencies of sounds.

4. A graphical user interface that communicates with the SGL to handle any or all the above, so that a lay user can easily adjust these settings without having to understand the underlying algorithms.

The above summary of the present invention is not intended to represent each embodiment, or every aspect, of the present invention. The detailed description and figures will describe many of the embodiments and aspects of the present invention.

DESCRIPTION OF PREFERRED EMBODIMENTS

As summarized above, embodiments of the invention provide a system and method for providing an improved user experience for 3D applications, such as software.

Customizing the SGL
Allowing customization of the internal rules the SGL uses to render sounds provides much greater flexibility to the overall user experience. For example, the user could change how much softer to play sounds that are further away.

Typically, a software application, such as a game, will send the positional information of a 3D sound to the SGL, and allow the SGL to determine how the physical sound itself is generated. Different SGLs by default handle this information differently. A different combination of API, sound driver and sound hardware may play sounds that are far away from a listener's position at completely different volumes.

For example, on a Creative® soundcard that implements both the DirectSound3D® and OpenAL APIs, if the SGL with the DirectSound3D® API is used, sounds that are far away are played more softly, or sometimes not at all, compared to the OpenAL SGL implementation. In this example, the DirectSound3D® SGL effectively makes a decision to "cut-off" sounds that have been arbitrarily decided as too far away for the listener to "hear," corresponding to a "maximum audible range." The reasons for this may be to reduce the number of sounds being played simultaneously (for purposes of reducing load on the sound hardware or other performance reasons), or for purposes of realism. In any event, this limit is arbitrarily decided in advance and is currently not adjustable.

A change made at the sound driver or hardware level of the SGL allows the user to decide at what "distance" the maximum audible range should be for sounds of various volumes. Therefore, if the software application sends the 3D sound data, the user may adjust whether or not this sound is actually played, and also the amount of volume drop off based on this distance. The specific algorithms to calculate this drop off are adjusted based on the user's input, so that the values used depend on the user's individual preferences, without necessarily exposing these algorithms to the lay user. This is particularly useful for game-critical sounds which are usually quite soft, such as footsteps that are far away, or silenced gunshots. In effect, this change allows a user to specify that, if the game provides that a sound should be available (by providing the sounds 3D data to the API), the user can "insist" on having it played by the SGL. Coupled with a GUI (such as, but not necessarily, the ones described in 4), this provides a lay user greater flexibility in how he or she receives 3D audio cues and
may provide a competitive advantage, in that he or she will be able to hear sounds at a
greater "range" compared to opponents with non-adjustable sound solutions that have sound ranges arbitrarily set low; adjustability has the added advantage of tweaking an optimum "range" since playing too many sounds at once may have a detrimental effect on the performance of the software application due to CPU load. This method can be combined with any of the other methods described here.

Amplification of Soft Sounds

The SGL may also be modified to allow amplification of sounds falling below a pre-set or user-set volume level, while leaving sounds that are above this level unmodified. This is analogous to a process known as normalization, used in DVD players where softer sounds, such as dialogue, are amplified so that they are audible without the watcher having to adjust his or her volume to the point where loud sounds such as explosions become too loud (or disturb the neighbors).

The inventive step here is allowing the user to tweak the amount of amplification to apply, and at what point this amplification should be done. Any and all sounds that fall below a preset or a user-set volume are amplified to another preset or user-set volume (e.g. all sounds that fall below -90db are amplified arbitrarily to -85db).

Ambient sounds (such as wind blowing, sound tracks in games) which typically are not handled as 3D sounds, could be selectively excluded from this selective amplification (as these sounds do not contain 3D data and are considered "traditional" sound sources), the advantage being that ambient sounds often do not contain critical information and may be distracting if they are amplified (i.e. the 3D footstep sounds are amplified, but the "ambient" sound of wind blowing is left quite soft). This method can be combined with any of the other methods described here.

Amplification of Selected Sounds

The SGL may also be modified so that it can identify specific sounds (which are effectively software files) and allows the user to selectively amplify these sounds, either on a scale (e.g., amplify all "gunshot.wav" sounds by 1.5 times) or a set minimum volume as described in 2 (e.g., amplify all "gunshot.wav" sounds to -85db if they fall below -90db). More generically, the SGL is modified so that specific frequencies, or ranges of frequencies have the above described rules applied, the effect being similar but with less granularity. The advantage here is that where the
software application has decided a particular sound should be very soft (e.g. a silenced pistol shot), a user can decide to play this sound at a much louder volume, which may provide him or her with a significant advantage over an opponent in a competitive environment.

The inventive step is both the user adjustability and the selective amplification of specific sounds (although this may not hold true for selective frequency amplification).

Again, as in 2, ambient sounds may be selectively excluded from this process. This method may be combined with any of the other methods described here.

**Graphical User Interface**

In a preferred embodiment, the foregoing are adjusted via a graphical user interface at the sound driver level of the SGL. Existing sound card drivers have graphical user interfaces which graphically represent speaker positions relative to the listener for the purpose of testing of speakers or positioning of speakers (e.g., simulating a speaker being closer, further away or at a different angle from the listener's position when it is not possible or convenient to adjust the actual physical speaker positions themselves due to physical constraints). This implementation differs in that this deals from those GUls in that, among other improvements, it includes control of the "range" in the virtual environment at which 3D sounds are generated by the SGL. Additionally, it may allow for sound-specific selections.

In one embodiment, the GUI includes a "radar"-like interface. The listener is graphically represented in the center of a ring that can be expanded or contracted graphically either by dragging on it with an input device such as a mouse, by means of a separate graphical slider, by inputting higher, or lower numbers or other similar methods as in 4b. The ring may represent several things:

i. The "distance" at which the arbitrary cut-off of soft sounds takes place as described in 1., by expanding or contracting the ring, the listener instructs the SGL to set this distance further or nearer the listener's relative position. This graphically shows that the user's listening range is "increased" or "decreased".

ii. The volume at which soft sounds should be amplified as described above. By expanding or contracting the ring, the user increases or decreases the level at which sounds which fall below a certain level are amplified.
This graphically conveys also the user's listening range as "increased" or "decreased".

iii. Both i and ii, and any other similar implementations are there to convey the notion of increasing or decreasing the user's listening range, much like how current video drivers allow the user to specify the distance at which objects may be viewed in a 3D environment (a shorter range improves performance since fewer objects are drawn, a greater range improves the amount that the user is able to see).

In yet another embodiment, the GUI includes a "counter" interface. In this embodiment, the GUI includes a box where distance may be described in virtual units in numerals (or no units at all) rather than a ring slider; e.g., the number of virtual "feet" that a listener is able to hear, with a similar implementation to (i) and (ii), above. This counter may be adjusted via a slider, or by typing in by the user, or any other similar input method. For example, it can be combined with the method as described above, so that as the ring is made larger, the numbers in the counter go up, and vice versa. This allows the user to correlate the numbers with the listening range/distance in the user's mind.

While the present invention has been described with reference to one or more particular embodiments, those skilled in the art will recognize that many changes may be made thereto without departing from the spirit and scope of the present invention. For example, many of the advantages of the invention have been discussed in the context of games. Other applications of the invention are readily apparent, such as simulations, movies and other audiovisual entertainment media, music and other audio, and the like.

Each of these embodiments and obvious variations thereof is contemplated as falling within the spirit and scope of the claimed invention, which is set forth in the following claims.
WHAT IS CLAIMED IS:

1. A customizable sound generation layer for a digital computing device comprising:
   an application program interface;
   a sound generation hardware; and
   a software driver for said sound generation hardware;
   wherein said software driver is adapted to be customizable by a user.

2. The sound generation layer of claim 1, wherein software driver comprises a set of parameters used by said sound generation layer to render sounds, and said customizable software driver is adapted to allow said user to modify said set of parameters.

3. The sound generation layer of claim 2, wherein said customizable software driver is adapted to allow said user to modify said set of parameters so as to modify a degree of amplitude reduction for a sound that is associated with a measure of distance.

4. The sound generation layer of claim 2, wherein said customizable software driver is adapted to allow said user to modify said set of parameters so as to select whether a certain sound is amplified.

5. The sound generation layer of claim 2, wherein said customizable software driver is adapted to allow said user to modify said set of parameters so as to select whether a certain sound is reduced in volume.

6. The sound generation layer of claim 2, wherein said customizable software driver is adapted to allow said user to modify said set of parameters so as to select whether a certain set of sounds is amplified.

7. The sound generation layer of claim 2, wherein said customizable software driver is adapted to allow said user to modify said set of parameters so as to select whether a certain set of sounds is reduced in volume.
8. The sound generation layer of claim 3, further comprising a graphical user interface adapted to enable said user modification.

9. The sound generation layer of claim 8, wherein such graphical user interface comprises a radar-type interface, and wherein said user is graphically represented in a center of a ring that can be expanded or contracted graphically using an input device.

10. The sound generation layer of claim 8, wherein such graphical user interface comprises a counter, and wherein said counter corresponds to a distance with respect to which said user desires to hear sounds within a range of volumes or a range of amplifications.

11. A method for providing an enhanced 3D audio experience to a user comprising:

   providing a customizable sound generation layer comprising:

   an application program interface;

   a sound generation hardware;

   a software driver for said sound generation hardware, wherein said software driver is adapted to be customizable by a user, said software driver comprising a set of parameters used by said sound generation layer to render sounds; and

   customizing said software driver based upon input from said user.

12. The method of claim 11, further comprising modifying said set of parameters based upon an input from said user.

13. The method of claim 12, wherein said modifying comprises modification to a degree of amplitude reduction for a sound that is associated with a measure of distance.

14. The method of claim 12, wherein said modifying comprises selection of whether a certain sound is amplified.
## INTERNATIONAL SEARCH REPORT

**International application No**: PCT/US2006/007880

### A. CLASSIFICATION OF SUBJECT MATTER

**INV. G06F 3/16**

### B. FIELDS SEARCHED

**Minimum documentation searched** (classification system followed by classification symbols)

G06F

**Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched**

### C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication where appropriate, of the relevant passages</th>
<th>Relevant to claim No</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>WO 01/50598 A (TRUE DIMENSIONAL SOUND), INC, SCARPINO, FRANK, A; GARCIA, ARTURO, H; BL 12 July 2001 (2001-07-12) figure 2 page 14, line 1 - line 18 page 17, line 4 - line 24 page 19, line 24 - page 20, line 6 page 21, line 29 - page 22, line 3 page 27, line 22 - page 28, line 25 page 37, line 10 - line 28 page 45, line 12 - page 46, line 3 page 51, line 18 - page 52, line 9</td>
<td>1-14</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-/-</td>
</tr>
</tbody>
</table>

**Further documents are listed in the continuation of Box C**

**X** See patent family annex

### Special categories of cited documents

- "A" document defining the general state of the art which is not considered to be of particular relevance
- "E" earlier document but published on or after the international filing date
- "L" document which may throw doubts on prior art claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed

### Date of the actual completion of the international search

1 August 2006

### Date of mailing of the international search report

18/08/2006
<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>figure 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>paragraph [0032]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>paragraph [0035]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>paragraph [0067] - paragraph [0069]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>paragraph [0075]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>paragraph [0138] - paragraph [0139]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>paragraph [0148]</td>
<td></td>
</tr>
<tr>
<td>Patent document cited in search report</td>
<td>Publication date</td>
<td>Patent family member(s)</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-----------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 2005091065 A1</td>
</tr>
</tbody>
</table>