EAGLE

ENVIRONMENTAL AUDIO GRAPHICAL LIBRARIAN EDITOR

VERSION 2.0

EAGLE TUTORIAL

CONTENTS

1 About the tutorial……………………………………………………….…………*page 4*
Who is the tutorial aimed at? What will the tutorial teach me? What hardware and software will I need?

2 Introducing EAGLE……………………………………………………..………*page 5*
So what is this EAGLE program anyway? How does it fit into my game's audio implementation?

2.1 Introducing the EAX extensions to Direct Sound
2.2 Introducing the EAGLE tool
2.3 Introducing the EAX manager

3 Installation……………………………………………………………….………*page 9*
How do I get the software provided up and running on my system?

3.1 Installing EAGLE
3.2 Installing the latest SB Live! drivers
3.3 Upgrading Unreal Tournament
3.4 Copying the tutorial files

4 Getting Started……………………………………………………….…………page 10
How does the EAGLE interface work? What features does EAGLE offer for enhancing my audio?

4.1 Starting EAGLE
4.2 EAGLE components and the Project Tree
4.3 Moveable Dialog Boxes in the EAGLE GUI
4.4 I can't see the Dialog Box I need– what do I do?
4.5 Slider properties

5 Working with Environments ……………………………..…………page 12
How can I use EAGLE to work with environments?

5.1 Environments within Unreal Tournament's *DM-Pressure* level
5.2 Creating a new Environment
5.3 Loading a sound source
5.4 Editing an Environment
5.5 More Environments

6 Working with Obstacle……………………………..…………………..…page 15
How can EAGLE help me to simulate sound absorption?

- **6.1** Obstacles within Unreal Tournament's *DM-Pressure* level
- **6.2** Creating a new Obstacle
- **6.3** Editing the Obstacle parameters

7 Working with the Geometry data………………..………………..…………page 17
How can EAGLE help me locate effects in different areas of my game geometry?

- **7.1** Importing geometry data
- **7.2** But my game's geometry format is not supported!
- **7.3** Navigating the geometry
- **7.4** Customising the Geometry Window
- **7.5** Environments, Partitions, and the cursor
- **7.6** Creating your first Partition
- **7.7** Adding your effects to the level
- **7.8** Oh no, the colour has 'leaked' out of the pressure chamber!
- **7.9** Finishing off the level

8 Sound Sources in EAGLE……………………………..…………………page 24
How can EAGLE help me to set up Direct Sound 3D buffer properties?

- **8.1** Sound Source features in EAGLE
- **8.2** Controlling the Sound Source
- **8.3** Placing sources within the geometry
- **8.4** Auditioning effects in the level

9 EAGLE Techniques…………………………………..…………………page 26
How can I get the best out of EAGLE?

- **9.1** EAGLE optimizations
- **9.2** 'Special' geometry data
- **9.3** Partition only BSP trees and the cube brush
- **9.4** Dealing with leaky maps

10 Conclusions………………………………………………………………page 28
How will EAGLE be maintained? Who should I contact for more information?

1. ABOUT THE TUTORIAL

In the last year, many developers have enhanced their PC game titles using 3D audio, mostly through Microsoft's Direct Sound 3D API. Creative Labs audio hardware has always provided advanced features for games, and the latest Sound Blaster Live! range is no exception. As well as supporting hardware accelerated mixing for Direct Sound 3D, the Sound Blaster Live! card has the ability to add some advanced audio effects, known as Environmental Audio. They are accessible to the programmer using an API called EAX – Environmental Audio eXtensions.

EAGLE is a tool that empowers the sound designer, with the ability to design, audition and archive different EAX audio effects. EAGLE also has the capability to associate effects with locations in a game's geometry. Working through this tutorial will give an **audio designer** a good grounding on exactly how to use EAGLE to enhance a game's sound with fantastic EAX effects.

An **audio programmer** will benefit from looking through the tutorial to see how EAGLE data can be incorporated into an audio engine. He would also be advised to read the documentation for EAX Manager DLL, supplied with EAGLE, which describes how this API retrieves EAGLE data from file.

Similarly, a **project manager** reading the tutorial can get a clear understanding of EAGLE's intended role within the project, and how the tool will aid both the designer and the programmer, and simplify their relationship.

Creative Labs worked closely with the developer responsible for the audio code in the games *Unreal* and *Unreal Tournament* when designing EAGLE. The audio implementation of *Unreal Tournament* includes calls to EAXManager.dll, making the title an ideal platform for testing EAGLE. For that reason, the geometry used in this tutorial is taken from an *UT* level. Of course, you'll need a fairly fast machine with a 3d accelerator and, most importantly, a SBLive! to run the editor and the game.

You will not need a copy of *UT* to run this tutorial, but you will need a copy of *UT* to fully playtest the environmental audio level representation generated in the tutorial. You may also find it handy to have a copy of *UT* simply to help you to visualise the geometry of the level we're going to work on, as the 3d rendering provided by EAGLE is fairly rudimentary (i.e. monochrome, no textures)

2. INTRODUCING EAGLE

2.1. Introducing the EAX extensions to Direct Sound 3D

Creative's Environmental Audio eXtensions (EAX) are Property Set extensions to Microsoft's Direct X API. They add 3D reverberation capabilities, and Occlusion and Obstruction effects to the Direct Sound 3D component of Direct X. (Note : EAX is *not* a replacement for Direct Sound – in fact, it actually relies on the 3D positional information for each sound source to be passed to the soundcard, using the appropriate functions in the Direct Sound API.)

Hearing Direct Sound 3D without these environmental audio effects, a listener can tell the direction each sound source is coming from, but has no idea of the environment where the sounds are located. For example, a sword fight taking place in a small padded cell should sound very different to a sword fight taking place in a huge cathedral.

Furthermore, Direct Sound 3D does not provide the audio effects necessary to simulate sound absorption by obstacles. Sound coming from behind a wall should be muffled in comparison to the same sound with a clear path to the listener.

EAX 2.0 provides the flexibility to create effects that simulate a vast number of different audio environments. It also allows, by applying occlusion and obstruction parameters, to correctly render sound sources that are for instance behind a pillar (obstructed) or in adjacent rooms (occluded).

For complete information about the EAX API please refer to the EAX 2.0 SDK which has been included on the EAGLE V2.0 CD.

2.2. Introducing the Environmental Audio Graphical Librarian Editor (EAGLE)

EAGLE is a tool for sound designers who want to be able to create realistic 3D aural worlds for their games. The GUI makes the process of designing and auditioning custom Environment Models (reverbs), Obstacle Behaviors (occlusions and obstructions) and parameter sets for Direct Sound 3D buffers (Sources), very intuitive. The EAGLE file format allows for easy archiving of effects and the EAX Manager DLL (see below) makes it simple for the programmer to access the designs.

The designer has the additional capability to assign an Environment to different areas within the level. He can also attach his Obstacle Models to the various walls that make up the areas in the 3D world. To do this, it is necessary for EAGLE to import a copy of the level's geometry. Currently EAGLE supports 3D Studio Max files (*.ase), Microsoft Direct X files (*.x) and Lightwave 3D Object files (*.lwo) but, via a Plug-in it is possible to write a custom file import routine and provides EAGLE with a raw list of polygons from your proprietary geometry format.

Once the level geometry has been imported, it is necessary to divide it into an appropriate number of acoustically different sections or *Subspaces*, using *Environment Partitions.* An Environment Partition is simply an invisible polygon whose only purpose is to 'plug' the gaps in the geometry (e.g. door openings) in order to separate each section. The resulting combination of geometry and Environment partitions will then be used to create an 'Audio BSP' tree that contains all the details about which

Environment and Obstacle models are to be used in each Subspace. This BSP tree, together with a list of the Environment and Obstacle Model parameters that have been used in the level, can be saved in an Environmental Audio Library file (.EAL).

2.3. Introducing the EAX Manager

Using the EAX Manager DLL (EAXMAN.DLL), the programmer can, at run-time, retrieve from one of these EAL files the EAX effects created by the designer.

The DLL will also take care of accessing the appropriate information from the Audio BSP, if desired. All that the programmer needs to do is, at regular intervals, tell EAXMAN the location of the listener (normally the player) and then each of the positions of the sound sources. EAXMAN will then return the details about which Reverb model should be switched on, and the appropriate Occlusion and/or Obstruction parameters which should be set for each individual emitter.

(Note, EAXMAN will not send any information directly to the soundcard – instead, it has been designed to allow the programmer to alter, if required, any of the parameters that have been retrieved from the .eal file before passing them to the soundcard via the Direct Sound EAX Property Sets interface)

3. INSTALLATION

3.1 Installing EAGLE

Install the EAGLE application using *setup.exe* from the *\EAGLE* directory on the CD-ROM. You *must* make sure you are using the latest SoundBlaster Live! Drivers (Nov '99 release) to use EAGLE correctly.

3.2 Installing the Sound Blaster Live! drivers

If you already have LiveWare 3.0 installed, ensure that you have the latest drivers by running the w9xdrv.exe in the *\Drivers\LW30Update* directory on the CD-ROM.

If you have an earlier version of LiveWare (1.0, 2.x), first upgrade to 3.0 by running the setup.exe in the *\Drivers\Liveware3* directory. Then update the drivers as described above.

3.3 Upgrading Unreal Tournament

If you want to quickly check out the EAX enhancements made in the tutorial within a game, you'll have to buy or blag a copy of Unreal Tournament. Install the game to your hard drive and download the official patch from Epic Website to upgrade to ver. 4.05.

Unreal Tournament will not be able to generate EAX effects if Aureal's A3DAPI.DLL library is present in your *Windows\System* directory – we recommend you search for and disable this file if you are attempting to test EAX with *UT.* To add EAX 2.0 support to Unreal, you need to copy the two files contained in the *\UnrealPatch* directory on the EAGLE CD (*Galaxy.dll*, and *Eaxman.dll*) to the *\UnrealTournament\System* directory on your hard drive. (You may like to backup the original version of *galaxy.dll* before copying these files to your hard drive)

3.4 Copying the tutorial files

Copy the contents (including subdirectories) of the *\TUTORIAL* directory on the CD-Rom to a convenient place on your hard drive.

4. GETTING STARTED

4.1 Starting EAGLE

Run the EAGLE.EXE from the start menu – by default it will be in the path *Programs\ Eagle*.

4.2 EAGLE components and the Project Tree

EAGLE allows you to work with different categories of EAX 2.0 effect to enhance the audio experience of your game. Look to the left-hand side of the screen, and you should see the Project Tree window:

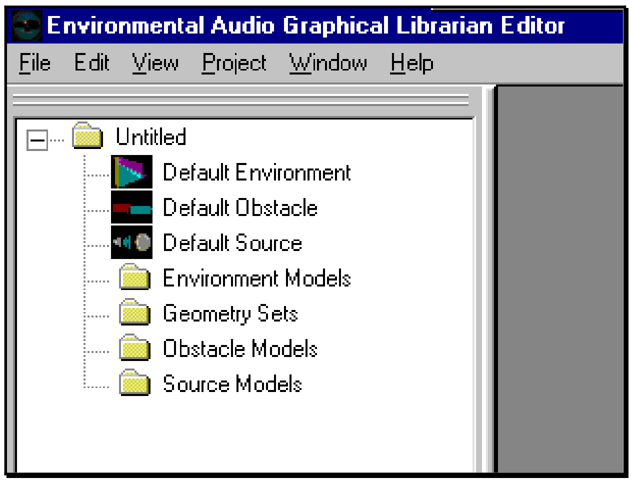

Figure 1 - The Project Tree window

Each component you design will be stored in the appropriate folder within the Project Tree. An Environment Model is a collection of settings which define an EAX2.0 reverb. This would typically be used to simulate a particular acoustic environment (See topic 2.1, *Introducing the EAX extensions to Direct Sound* and section 5, *Working with Environments*).

An Obstacle Behavior Model is a collection of settings which could be applied to a Direct Sound 3D buffer. This is use to simulate the sound absorbtion that occurs when an obstacle is blocking the direct path of a sound source. (See topic 2.1, *Introducing the EAX extensions to Direct Sound* and section 6, *Working with Obstacles*).

Source Models store Direct Sound 3D parameters, which can be applied at initialisation time to 3D buffers in order to control their directivity and loudness, and therefore how far away they can be heard. (See section 8, *Working with Sound Sources)*.

Geometry Sets come into play when you want to relate effects you have created to areas within a level geometry. Each set of geometry data loaded into EAGLE will be stored in the Geometry Sets folder. To keep the EAGLE data files down in size, you may wish to create separate projects for each level of your game.

4.3 Moveable Dialog Boxes in the EAGLE GUI

The dialog boxes on the right hand side of the screen provide you with an interface to adjust the parameters for the currently selected instance of each component. As well as a box for each component type, there is a box for manipulating EAGLE's geometry brush. These dialog boxes are moveable and dock-able, so you can personalise the workspace just how you want it. This goes for the Project Tree window also. The window positioning is persistent, so when you re-open EAGLE they will re-appear as they were when the application was last closed down.

4.4 I can't see the Dialog Box I need – where's it gone?

If you can't find a particular dialog you need, its probably been closed down. Just click on *View, Control Windows…* to bring up a dialog that allows you to toggle the visibility of each of the dialog boxes.

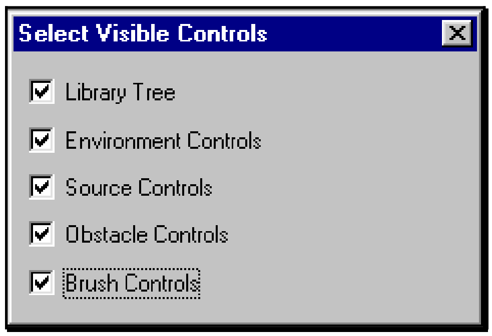

Figure 2 - Selecting the visible Control panels

4.5 Slider properties

Each EAX 2.0 / Direct Sound 3D parameter available to the designer in EAGLE is adjustable using sliders. It is possible to change the sensitivity of each slider by changing its proprieties. Right click on any slider and select *Properties. A dialog box will appear allowing the user to change the units per step and how many steps the slider will move for each Page Up or Page Down operation.*

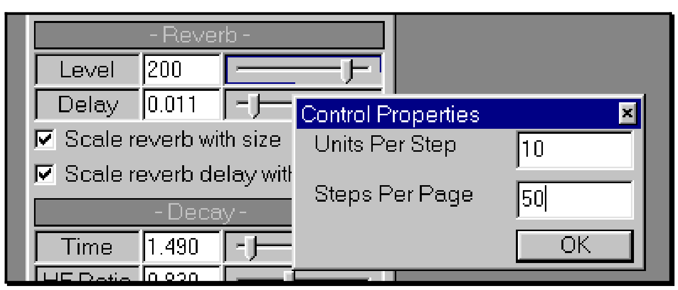

Figure 3 - Adjusting the slider resolution

5. WORKING WITH ENVIRONMENTS

5.1 Environments within Unreal Tournament's *DM-Pressure* level

Now you know your way around EAGLE's interface, it's time to start working on some effects. We're going to be using EAGLE to enhance the audio in a level from Epic's *Unreal Tournament* – a deathmatch map called *DM-Pressure*. If you have a copy of *UT*, now might be a good time to check that level out. Load up the game and start a practice deathmatch in the *DM-Pressure* level.

Figure 4 – Different halls in the *DM-Pressure* level

Run round and check out the level. You will notice that the *Unreal Tournament DM-Pressure* level consists of three main halls, one at the east, one in the centre and one to the west, connected by several passageways. Some of the corridors are on the walkway level and some on ground level; one of the passages is filled with water. Using reverberation to simulate environments is a key enhancement offered by EAX, so now lets simulate some of the environments present in *DM-Pressure.*

5.2 Creating a new Environment

The three main halls make up the largest proportion of the space in the *DM-Pressure* level, so first of all we'll set up a reverb to suit the acoustics of these halls.
To create a blank new environment, right click on the *Environment Models* folder in the Project Tree and select *New*.

A new Model appears in the folder with the name *Environment1*. A 3d representation of the 'shape' of the reflection / reverberation combination is displayed. For more information about this, see the *Environment Model Window* section of the help file. You'll need the *Environment Model Controls* panel to adjust the EAX2.0 parameters – once again try *View, Control Windows…* if this control is not present (see topic 4.4).

5.3 Loading a sound source

When editing a reverb it is useful to play a sound for reference. Right click on *Source Models* within the Project Tree and select *New*. An open file dialog box appears allowing you to search for a WAV file. Find something relevant to the game, such as a sample of footsteps or gunfire, which will sound good when looped.

The Source Model window that appears allows you to play and stop the sample, and to change its 3D position and orientation. For more information, see *Source Model Window* in the help system and Section 8, *Working with Sound Sources*. Click the play button on your new source's window to set it playing – the reverb currently being edited will effect the sound so you can tell roughly what effect the reverb will have on the game.

5.4 Editing the reverb parameters

The Combo Box named Preset contains the 26 Environments inherited from EAX 1.0, and give the EAX 2.0 designer a good starting point for developing a new Environment Model. The first step towards getting the effect we're looking for is to select the Hangar Environment. Notice that the parameters change to show the values that make up the Hangar reverb. To simulate the walkways, pipes and pillars within the level, bring down the *diffusion* parameter to 0.9, breaking up the reverberation a little. The initial reflections could be more prominent – try setting the *Reflection Level* to about –350, and increasing the *Reflection Delay* a little to make the space sound larger.

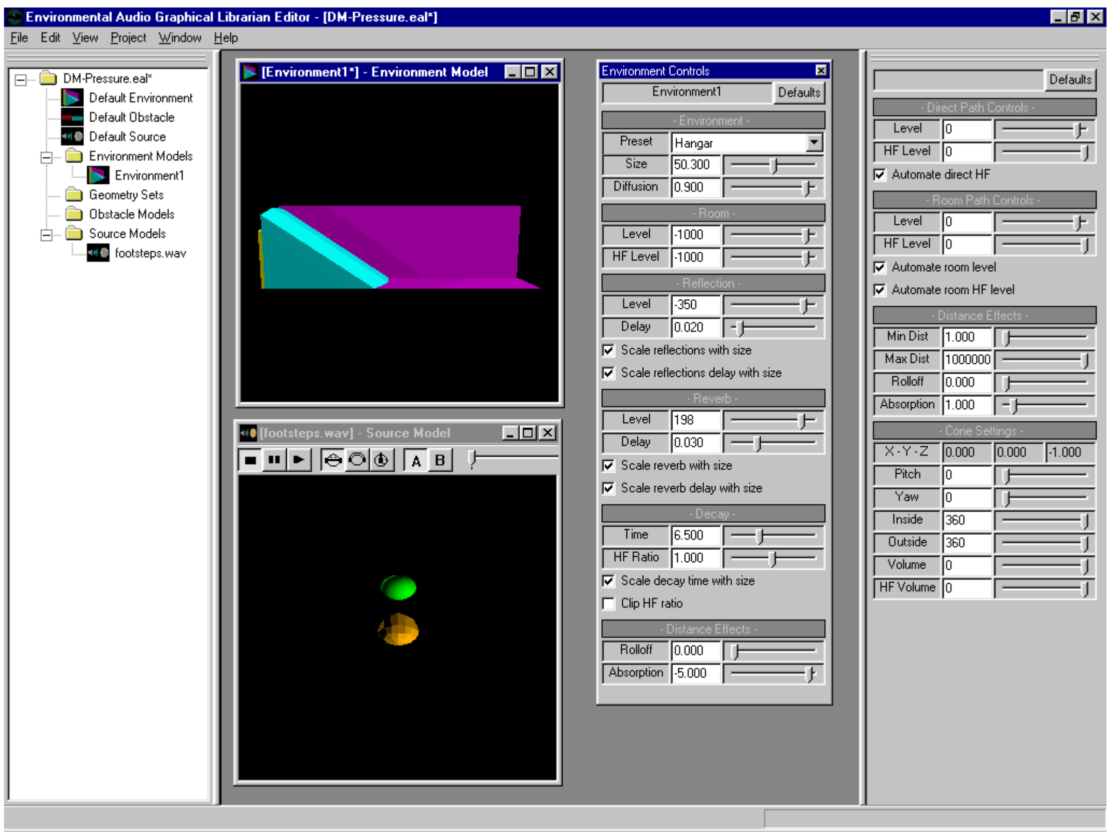

Figure 5 - Editing an environment

To emphasise the reverberant nature of the room, increase the *Reverb Level*, and change the *Decay time* to about 6.5. Try raising the presence of the high frequencies within the reverb by increasing the *Decay HF Ratio* to around 1.0 (untick the *Clip HF ratio* box if necessary).

5.5 Naming the Environment

Your new Environment's name property is very important, as the programmer, via the EAX Manager API, can query for Environment Model data by name. Right click the new environment's entry in the *Environment Models* folder, and select *Properties*. Change the name to something meaningful, such as *Main Hall*. When you close down the Environment's graphic display window, you will be asked whether you want to update the environment. If you are happy with the reverb you've made, click *Yes* and the new Model will be stored in the Project Tree. To open an Environment Model for editing, simply double click on its name from the project tree. This will open its Window and, if not already open, the Environment controls panel too.

5.6 More Environments

When I was playing this level, I identified four different basic Environment types. We have just worked on the halls' reverb. I also found passageways, waterways and the pressure room. Depending on just how much work you want to do, you could further differentiate between, for example, the different length corridors – but we'll keep things simple now.

Create another new environment and use the *Environment Controls Panel* to whip up an effect for the passageways (I used *Bathroom* preset to start me off – use whatever works for you!) Continue to create environments in the same way for the *Waterways* (I suggest the *Under Water* preset as a starting point) and the *Pressure room* (Maybe something a bit off the wall here – this room is a *special* in the level).

If you want to check out and maybe utilise some of our reverb designs that ship with this tutorial, right click the Environment Models folder in the Project Tree and select *Import*. Choose the *DM-Pressure Environment.eal* data file from the EAGLE tutorial folder on your hard drive or CD, and highlight the remaining reverbs we need for the level. Click OK and they will appear in the Environment Models folder. You can audition, edit and use them just like your own reverb designs.

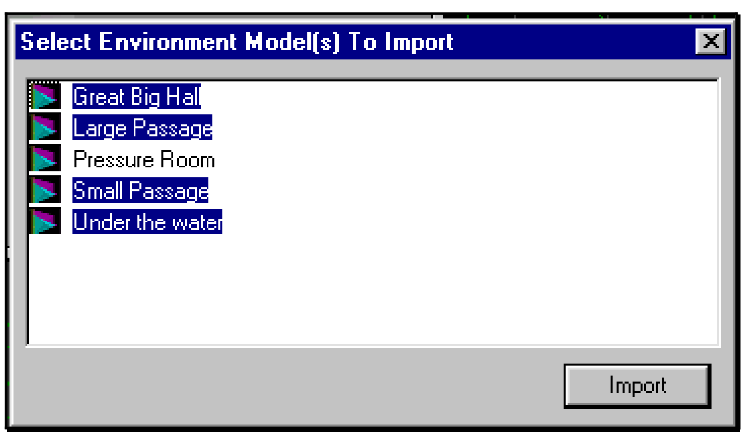

Figure 6 - Importing more Environments

Similarly, if you want to archive your own reverb designs for use in future projects, it is simple to create EAGLE data files containing just a load of your own Environment Models, which can be imported into other projects in the same manner.

6. WORKING WITH OBSTACLES

6.1. Obstacles within *Unreal Tournament's DM-Pressure* level

As described in topic 4.2, an Obstacle Behavior Model is a set of EAX 2.0 parameters which can be applied to a Direct Sound 3D buffer to muffle the sound, simulating obstruction or occlusion. In the *DM-Pressure* level, the main structures that are likely to come between the listener and a sound source are the walls. These appear to be made of solid stone blocks, and we're now going to create an Obstacle Model to simulate the acoustic behaviour of these walls. This will allow us to introduce occlusion parameters to describe how much the walls might absorb sounds passing through them.

6.2. Creating a new Obstacle

Right click on the *Obstacle Models* folder in the Project Tree and select *New*. A new Obstacle Model called *Obstacle1* will be created, and a graphical representation of the effect will be displayed. Locate the *Obstacle Controls* panel, which will be used to adjust the Obstacle parameters.

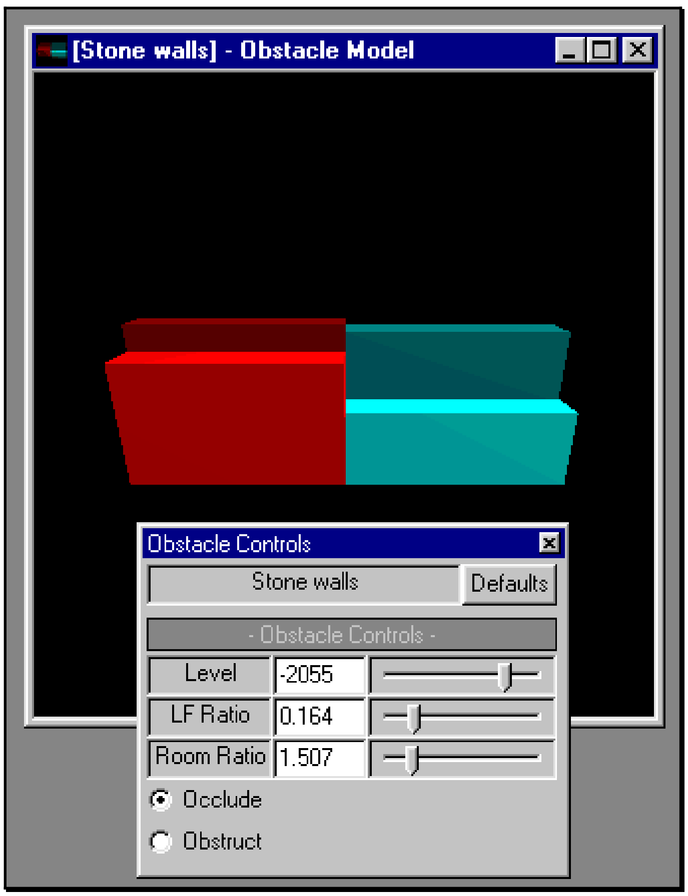

Figure 7 - Editing an Obstacle Model

Once again, it will be very useful to play a looping sound whilst editing, to audition the obstacle settings (see Topic 5.3). The Obstacle parameters are quite simple.

6.3. Editing the Obstacle parameters

The *Level* sets the overall amount of muffling (a combination of filtering and attenuation) applied to the sound. The thick stone walls would muffle the sound considerably – set the level to around –2000. *LF Ratio* allows the designer to change the spectral quality of the sound by biasing the absorption towards high or low frequencies. Stone walls will absorb a lot of high and some mid frequencies – set this value to around 0.16.

When the *Occlude* radio button is selected, the *Room* parameter is available, allowing the designer to apply additional muffling to the reverberated portion of the sound. These walls will be used to occlude sound, so make sure the O*cclude* box is selected. The default value for the *Room* parameter is very reasonable for most situations. It goes without saying that you'll want to tweak the slider and see how the sound is altered, but the default value will do very well for our stone walls.

When your Obstacle Model is ready, you can name it and save it in the project tree. Double clicking on an Obstacle Model from the project tree will open the Model Window and Controls panel. And of course you can use the librarian features of EAGLE to archive Obstacle Models and import them into different projects. So if you wish, you can import the remaining Obstacle Models we will need from the *DM-Pressure Obstacles.eal* file provided with this tutorial.

7. WORKING WITH GEOMETRY DATA

7.1. Importing geometry data

To associate the effects that we've created with different locations in the level, we need to work with some game geometry. Right click on the *Geometry Sets* folder and select *Import*. The *Import Geometry* dialog will be displayed, allowing you to first select a geometry format, and then search for and load some geometry data.

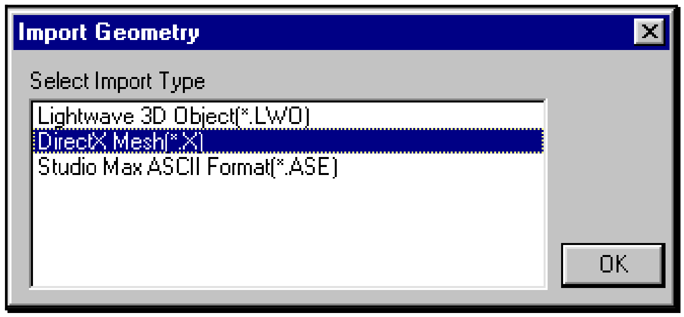

Figure 8 - Import geometry dialog

'Out of the box', EAGLE can understand a number of different geometry formats including Direct 3D's X file format, as well as formats from popular 3D design packages such as 3D Studio Max and LightWave. However, if the geometry data you need to load is not in one of these formats, don't despair.

7.2 But my game's geometry format is not supported!

Firstly, investigate the possibility of converting the geometry to an EAGLE supported format. If this is not possible, ask your 3D programmer to look at the EAGLE geometry plug in SDK. This will involve him writing an Import Component that can be used by EAGLE to import your specific geometry file format. Import Components can be registered with the application to make them available from the Import Geometry dialog box as standard file types.

7.3 Navigating the geometry

We'll need to load the *DM-Pressure* geometry, which is provided for you in .X file format. Make sure that the file type *DirectX Mesh (*.X)* is highlighted. Click *OK* and locate *DM-Pressure.X* on the CD or your hard drive (see topic 3.4, installation). Now, you may like to rename the geometry set you just loaded. Open the *Geometry Sets* folder in the project tree, right click on *Geometry1*, select properties and give the geometry a more meaningful name, such as *DM-Pressure geometry*. It could also now be helpful to maximise the 3d-view window at this point.

The 3d edit view shown in Figure 9, which should be familiar to anyone who has spent much time working with a 3d level editor, includes four quadrants, and is described in the section *Geometry Graphic View* in EAGLE's help system.

At the top left is a simple, monochrome rendering of the scene. The camera can be moved and its direction altered by holding down various combinations of the mouse buttons and dragging in this quadrant. It is also possible to navigate the 3D View using the cursor key in combination with the mouse. For more information on navigating round the geometry, see *Moving the Viewer in Geometry Rendered View Modes* in the EAGLE help system.

Buttons	Drag axis	Result
Left	X	Change orientation left / right
Left	Y	Move forward / backwards in current orientation
Both	X	Slide (strafe) left / right
Both	Y	Slide up / down
Right	X	Change orientation left / right
Right	Y	Change orientation up / down

Table 1 - Mouse movement in rendered 3d view

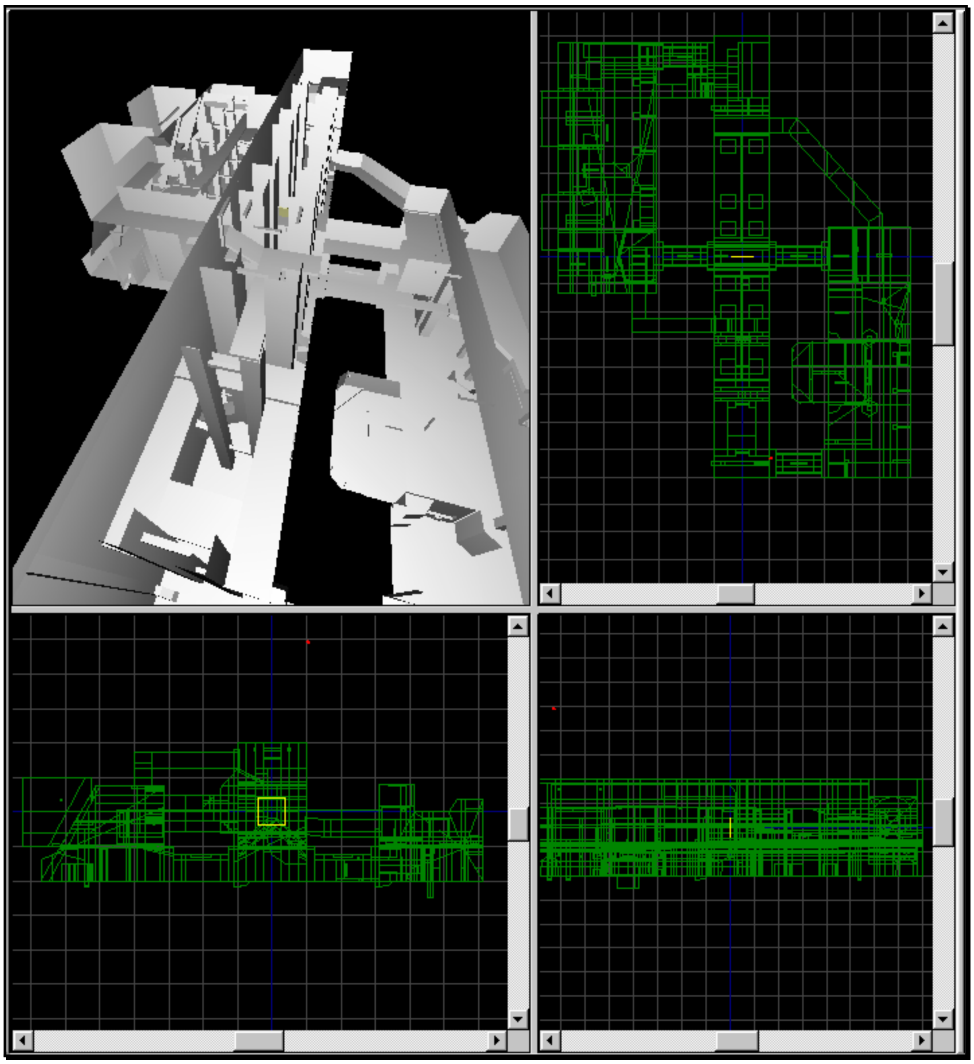

Figure 9 - The Geometry window

The top right quadrant shows a wireframe *plan*, or top down view of the geometry. The bottom right quadrant shows a wireframe view looking along the East/West axis, the bottom left along a North/South Axis. The small red dot is actually an arrow showing the position and orientation of the camera in each view, and the yellow square / line shows the current shape / size / orientation of the portal cursor. The red arrow orientation and position are updated in real time based on the position of the camera in the 3D View.

7.4 Customising the Geometry Window

Depending on your geometry scale, you might want to change the default settings for the Geometry Windows. To do this, from the *View* menu, select *Edit Window* and then *Geometry….*This will open a dialog box where you can change different parameters such as Grid and icon size, mouse and keyboard sensitivity.

7.5 Environments, Partitions and the Cursor

OK, theory time. The EAX manager (see topic 2.3) uses its own representation of a level's geometry. The idea behind EAGLE 3D editing is to separate this geometry into discrete areas, each with an associated EAX 2.0 Environment. As an audio designer, you will most likely want to distribute a different Environment to each area present in the level.

The walls, ceilings and floors described by the level's geometry help already to define the areas, and EAGLE will use them to help contain an Environment. However, there are gaps, doors and windows, which exist to allow the player to move between rooms. For the purposes of EAGLE's EAX management, each of these openings must be 'plugged'. In other words, to assign a particular Audio Environment to a specific area, this must be fully bounded by a combination of the level designer's wall and floor polygons, and the EAGLE editor's *Environment Partitions.* These areas are called *Subspaces* within EAGLE.

For example, look at the pressure chamber in the east hall of the geometry. For those unable to use *UT* to view the level, here is a picture:

Figure 10 - The pressure chamber in the east hall

The pressure chamber is bounded by walls, a roof and a floor, but there is a door to let the player enter and exit. As an audio designer, you may decide that a player inside the pressure chamber should hear audio processed using a different reverb to when he is outside the chamber. So you must seal the door opening using an Environment

Partition, EAGLE's invisible polygon, and make the chamber a Subspace – here's how…

7.6 Creating your first Partition

Navigate through the level using the 3d rendered view until you find the pressure chamber (it will be in the east hall). The screen should look something like figure11. Notice, the three 2d views have the red cursor in their centres – to achieve this, right click within each quadrant and select *centre on view*. You should also note that left dragging up and down in the quadrants sets the zoom level for each.

Unfortunately, as you can see from the bottom two projections, the door space extends down below the rest of the level to accommodate the pressure chamber's vertically sliding door. No matter, we'll just have to make a big Environment Partition to cover the whole gap.

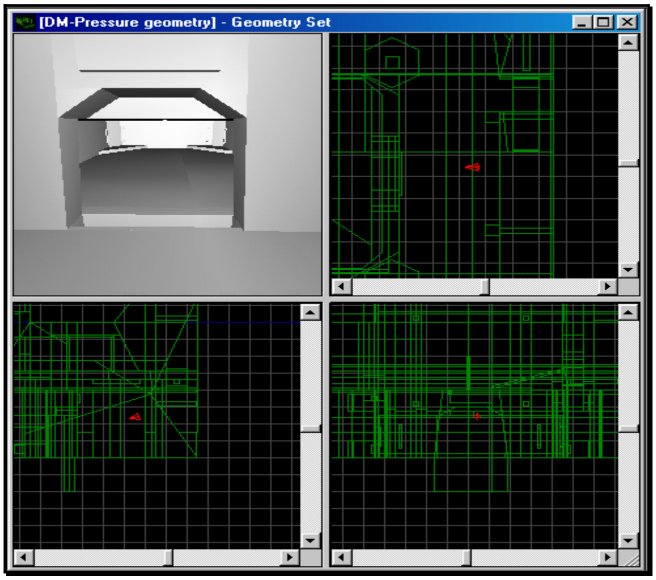

Figure 11 – The cursor covers the Pressure chamber doorway

Now right click in the plan view (top right quadrant), just about where the doorway gap is (where the red camera arrow points) and select *position cursor*. The yellow cursor line will appear, probably along the east/west axis. If the Partition is on this axis, it will be no use blocking off the door gap, so find the *Brush Control panel* (*View, Control Windows… Brush Controls* if its not present*)* and set the *yaw* parameter to 90 degrees.

If all is going well, you should now see a yellow transparent sheet somewhere in the proximity of the door gap in the rendered view. This means the cursor is in the correct orientation and nearby to where it is needed. All that remains is to adjust the size and exact positioning.

Drag the yellow line in the plan view with the shift key and the left mouse button held in order to get it into position in the doorway. In the bottom right quadrant, the cursor will show as a rectangle. The doorway should be easily recognisable. Once again hold shift and the left mouse button and drag the cursor until it's roughly in the middle of the doorway. Then use shift and both mouse buttons. You will see that moving the mouse in different directions grows and shrinks the cursor in two axes.

Buttons	Drag axis	Result
Shift + Left	X	Move cursor left / right in current projection
Shift + Left	Y	Move cursor up / down in current projection
Shift + Both	X	Make cursor wider / thinner
Shift + Both	Y	Make cursor taller / shorter
Shift + Right	X	Change cursor's orientation round current projection's z-axis
Shift + Right	Y	Change cursor's orientation round current projection's x-axis

Table 2 - Mouse controls for shaping and sizing the portal cursor

Work with these controls until the cursor covers the doorway's gap. The screen should look something like figure12.

It is important that the cursor covers the gap completely, right down to the bottom of the door space. The best technique to check this is a combination of moving the camera to view each intersection between partition and level geometry, and checking that the cursor on the 2d views always overlaps the relevant level polygons that it is adjacent to.

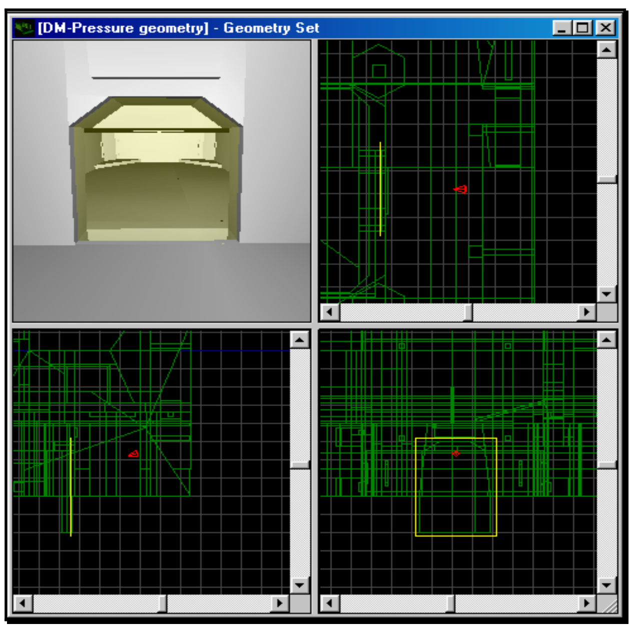

Figure 12 - cursor covers pressure chamber door

To add the Environment Partition select *Add Brush* from the *Project* menu (or press Ctrl+D). The yellow Partition cursor will appear to have turned red. This means that a Partition has been created. If the Partition is positioned correctly, the pressure chamber should now be a Subspace within EAGLE's geometry representation. To check this is the case, its time to start placing the effects we created earlier.

7.7 Adding your effects to the level

Before effects can be placed in the geometry, you must tell EAGLE to update the audio BSP with the Environment Partitions you've added. Hit *F7* or click *Project, Build*

Audio BSP… The settings for this dialog are explained in *Geometry Build* within EAGLE's help system. For now, just click OK. It's worth bearing in mind this process can take a while, especially on a level with complex geometry. It is best therefore to add a number of Partitions in between BSP tree builds to cut down on wasted time. Section 9 discusses BSP tree issues further.

Once the BSP tree has been built, return to the geometry view. You should still be looking at the doorway to the pressure room, with a Partition in place. Use the rendered view to move the camera well into the pressure room. Centre the 2D views and right click somewhere well inside the pressure room on the plan view. Select *Add Environment Tag*. Change the *Environment* to the *Pressure room* effect you just created, select the Occlusion property from the list of Obstacle Models and specify the name you want to give to this subspace in the specific field. EAGLE will automatically generate a new colour for each Environment Model available but, if you feel like it, you can set the colour to something you might associate with this effect For more information on the Environment Tag properties dialog box please refer to EAGLE on-line help.

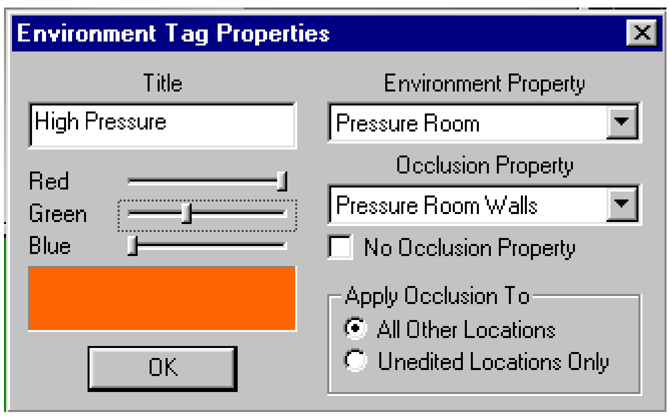

Figure 13 - Adding an environment tag

When you click *OK*, an icon will appear in the pressure room within the 3d rendering should become shaded in the colour you selected, like this :

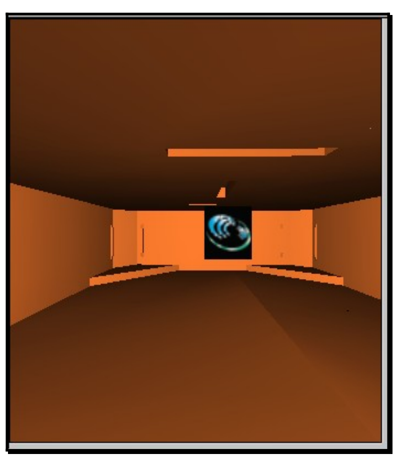

Figure 14 - New environment is shaded orange

Now move backwards within the rendered view. When you exit the pressure chamber, through the Environment Partition, you should see that the rest of the world is still monochrome black and white. This is good, because it means that there are no leaks – your portal has ‘plugged’ the doorway gap correctly. Notice also on the left side of the status bar the name of the subspace you are in. This is quite handy together with the colour scheme for quickly checking the ‘integrity’ of each subspace.

7.8 Oh no, the colour has ‘leaked’ out of the pressure chamber!

If this has happened, the Partition you created didn’t ‘plug’ the gap – Doh ! Don’t panic – just try again. You’ll see the Partitions on the 2D views as purple lines with purple squares in the centre. Locate the Partition in the pressure chamber doorway (this should be the only one present right now, if you’ve been following instructions!). Right click the purple square and select *Remove Partition*. Now go back to the section on **Creating your first Partition** and try again – we promise it’s possible! When you have rebuilt the audio BSP with the Partition correctly in place the colour will stop ‘leaking’, so you’ll know you have done it right.

7.9 Finishing off the level

If you want to practise using EAGLE, here’s some more work that needs doing on the *DM-Pressure* level. There are several corridors that join the different halls in the level. These include one passageway between the east and west halls positioned very high up. This passage runs between two square tanks, and in the game the tanks and the passageway are filled with water.

Polygons exist on the top of each tank to mark the water level, and these conveniently seal off the entire water based environment. However, with the normal passageways we are not so lucky, so you’ll have to add Environment Partitions at either end of each passage. When you’ve added the necessary Partitions, build the audio BSP (*F7*) and then go to the centre of each passageway and add environment tags referring to the appropriate effects. Don’t forget to check for ‘leakage’ caused by missing or inaccurate Partitions.

8. SOUND SOURCES IN EAGLE

8.1 Sound Source features in EAGLE

You have already used EAGLE's Source Model component to help audition audio effects, but there are several further parameters which you can adjust to help provide balance, range and directivity to Direct Sound 3D buffers within your game. For example, imagine a sound source in a game such as police siren. It's not relevant in this example whether its stationary or moving. However, directivity is important – you'll perceive a louder sound if you are in front of the siren's speaker than behind. Audible range will also be important – it could be crucial to the player to hear the approaching police car at a certain range.

EAGLE allows the designer to experiment with the Direct Sound's Min and Max Distance parameters, giving you the ability to determine the audible range of a sound buffer within the game world. Likewise, the sound's Cone parameters provide control over a sound's directivity. See the Direct Sound 3D documentation for a full explanation of these parameters, which are normally under-used in game audio implementations.

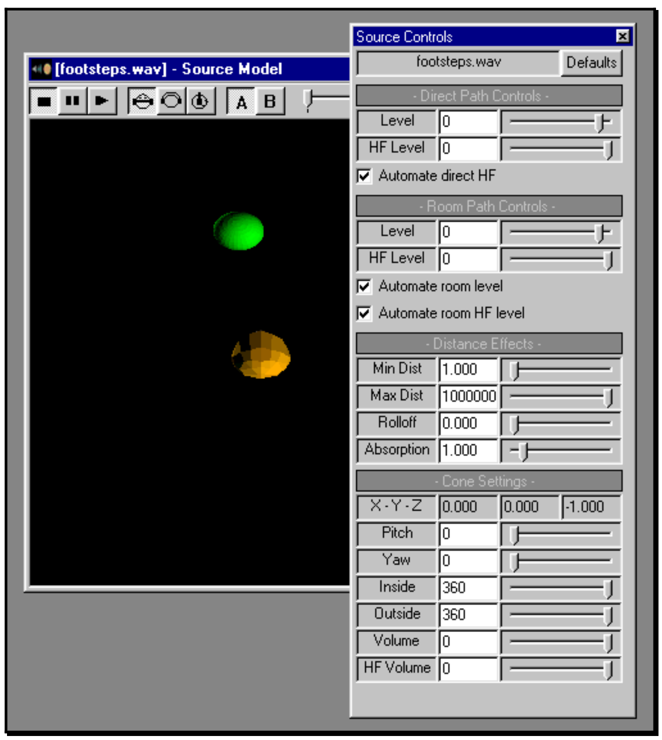

Figure 15 - Editing a source model

8.2 Controlling the source sound

A Source Model is independent of the .WAV file that you loaded when you created the source. In effect, the sound being played is just there to audition the Direct Sound 3D settings. However, you would expect that when you are adjusting the 3D settings for a particular sound in your game, you'll be using the same WAV data in auditioning. There are some cool features that this auditioning capability exposes.

Naturally, EAGLE renders its audio via Direct Sound 3D. The Source Model Window shown in Figure 15 allows the designer to audition the Direct Sound 3D settings, by controlling the 3D position of a sound buffer. The relative distance between the source, represented by the green sphere, and the listener, the 3D modelled head, is displayed on the status bar.

Once again, the Source Model name property is very important, because the programmer will query for Source Model by name, applying the retrieved data to a Direct Sound 3D buffer of wave data within the application.

8.3 Placing Sound Sources within the geometry

EAGLE can also be used to get a good idea of how a sound will come across during gameplay. If you are using geometry in your EAGLE project, you will have already covered placing environment tags into the map. Sound sources can be similarly placed. Open the *DM-Pressure* geometry set so you can see the level displayed graphically. Create a new Source Model by loading in the .WAV file *trickle1.wav*. This contains the water trickling sound sample that can be heard by the water tanks in the level.

Locate one of the two water tanks in the level geometry via the 3D rendered view. We're going to place the static, looped trickling sound just on the surface of the water, where it is audible in the game. Manoeuvre the camera to such a position, and right click one of the 2D views, where you want the sound to originate. Select *Add Source Tag*, and choose the *trickle1* from the list. A speaker icon will appear at the chosen position in each quadrant. Right click that icon and ensure that *mute source* is not ticked (right click on the icon for the context menu). You should be able to hear the trickling sound.

8.4 Auditioning effects in the level

Now try moving the camera around the level. Any effects that you have associated to subspaces in the geometry will be applied to the trickling sound. When you change from one subspace to another, the correct reverb will effect the source. Likewise, any obstacle model that has been specified will muffle the sound. The sound will become attenuated as the camera moves further away.

If you make changes to the trickle Source Model, altering the directivity or range of the sound, these will be audible also. Although we don't claim that EAGLE will accurately replicate the audio experience of your game (for instance, EAGLE has no notion of moving sounds), you should get a pretty good idea of how the different reverbs, obstacles and sources will interact.

9. EAGLE TECHNIQUES

9.1 EAGLE optimisations

If you've read this far, you're probably well aware that using EAGLE / EAX Manager techniques in your game, particularly when geometry processing is utilised, will involve a 'pay-off' situation. The choice of a BSP as the data structure used by EAGLE to separate geometry into subspaces means that at run-time, the search to identify the listener's or sources' subspace is very rapid. However, as geometry becomes more complex, build time and data size can increase. Obviously, then, the trick is to keep the geometry simple. We suggest a couple of techniques to achieve this aim.

9.2 'Special' geometry data

The level designer could create a special 'low polygon model' of each level for audio effect design. This could provide all the essential polygons that demonstrate the form of the design, but leave out all the intricacies of the world that will serve no purpose as far as audio goes. The level designer could help out another way – by including the notion of invisible polygons to zone off the rooms in the original model. This will save the audio designer the job of adding the Environment partitions, so only one BSP tree build would be necessary.

9.3 Partition only BSP trees, and the cube brush

One of the options on the *Build BSP tree* dialog is *Partitions Only*. When this is ticked, EAGLE will ignore all polygons in the geometry except for those added as Environment Partitions in EAGLE. This means that the EAGLE designer has to be responsible for completely enclosing each Subspace with Partitions. However, the complex world geometry will be ignored during the build, quickly producing a much smaller structure.

Section 7, *Working with the geometry* introduced the Partition Brush as a plane. However, the option exists on the Environment Partition control dialog to create cube, cylinder and sphere Partitions. This in fact means multiple Partitions can be created at once in a solid shape. So a good method for creating a small, efficient BSP tree would be this: Load the full geometry data for a level; Create all the subspace you need using cube Partitions for instance (the geometry you loaded will still be displayed, and will provide you with a map around which to design the subspaces); Then build the tree with *Partitions Only* set.

This method would be particularly suited to games set outdoors, maybe including racing games, where the geometry of the level does not necessarily dictate the distribution of different audio environments.

9.4 Dealing with leaky maps

Topic 7.7 introduced the problem of 'leaking' zones. As well as being caused by poorly placed Environment Partitions, this can be the result of level designs where polygons simply don't always join correctly – EAGLE will find the slightest 'hole' when the BSP tree is built. A solution is to change the plane thickness in the Build geometry dialog to a more appropriate value. As a rule of thumb you should use a value equal to the units per step used by the translation offset co-ordinates for the brush (see topic 4.5 – Slider

Proprieties). Cube Partitions can also come in handy in this situation, if the level designer cannot be persuaded to patch up his creations!

10. CONCLUSIONS

EAGLE is an ongoing project, and will be updated whenever the EAX SDK is improved upon, to give you instant access to the cool new features. As such, Creative positively welcomes any feedback you can give, so that our resources are put towards fulfilling you needs.

The EAGLE development team is very keen to hear your comments on how well EAGLE works for you and how the tool can be improved in future releases. Please do not hesitate to get in touch with Developer Relations in the Europe or the US (see below) with any criticism, constructive or otherwise!

We are keen to hear your feedback, and help you out in the unlikely event of things going wrong.

European Developer Relations
Creative Labs (UK)
Unit 2, The Pavilions
Ruscombe Business Park
Reading, Berkshire
RG10 9NN

Peter Harrison
+44 (0)118 982 8237
peter.harrison@europe.creative.com

Dan Peacock
+44 (0)118 982 8263
daniel.peacock@europe.creative.com

US Developer Relations
Creative Labs Inc
1901 McCarthy Blvd
Milpitas
CA 95035

Liam Byrne
408 546 6452
lbyrne@creative.com

Andrew Preston
408 428 6600
apreston@creative.com

http://developer.soundblaster.com

http://www.creative.com

http://www.europe.creative.com