



Oldunreal Multimediaupdate for Unreal and UnrealTournament by www.oldunreal.com

If you like those updates you can visit www.oldunreal.com and help me to keep it alive with a donation, its not easy for me nowadays, and with a little bit help you can help me improve these drivers, to buy new hardware for testing purposes, and it allows me to spend more time for it and the page itself.

This package is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. It contains an enhanced OpenGL Renderer (Videodriver) and the two new Sounddrivers OpenAL and FMod for Unreal Version 224/225(Sound only)/226 and RTNP/Unreal Gold, and UnrealTournament 436/351.

These updates are made by Oldunreal, and are completely free. The package may be redistributed without the author's prior permission, and must remain unmodified, but if you offer them for download somewhere please refer to my page. This package can be offered for download everywhere as long it is of no charge. This means especially those filesharing pages in which you have to register (and maybe pay) to get it. If you can't offer it for free don't put it on your page.

The license for OpenAL (www.openal.org, made by Creative Labs, Inc, www.creativelabs.com) and the license for Fmod (FMOD Sound System, copyright © Firelight Technologies Pty, Ltd., 1994-2005, www.fmod.org) which are used and shipped with these patches forbid a commercial use in any way too.

Special Thanks:

Epic (of course for sources,help and creating this game)

CreativeLabs for OpenAL and very fast response time in the mailing lists

FMOD because of leaving it free for non commercial use

Chris Dohnal for his great work on OpenGL and allowing me to use it (<http://cwdothnal.home.mindspring.com/utglr>)

All friends of Oldunreal who provided me with informations and time, helped me to find bugs and fixes (especially Shambler), and the donations i got so far.

And the most important friend who helped me to make this possible in hours and hours: Kerilk

Installation:

UPDATE YOUR WINDOWS TO LATEST DIRECTX!

Use the installer for your preferred base version of Unreal. Supported are the versions 224,226 and Unreal Gold. There is a version of the Audio parts for 225 too, but be aware that the 225 Version is based on a hack, so if it isn't working correctly there are almost no ways to fix it. I leave it as „unsupported“. The Unreal Tournament Version works with 4.36 and 4.51-maybe with others as well, only tested these two.

The installer is moving the original Unreal files which need to be replaced into a subfolder „backup“ within „Unreal/System“. The uninstaller provided with this package will restore every file to its original location.

The files to patch Unreal to 224/225/226/RTNP and UT to the latest version can be found on www.uldunreal.com or on the official sites. RTNP with the latest patch equals (technically) UnrealGold.

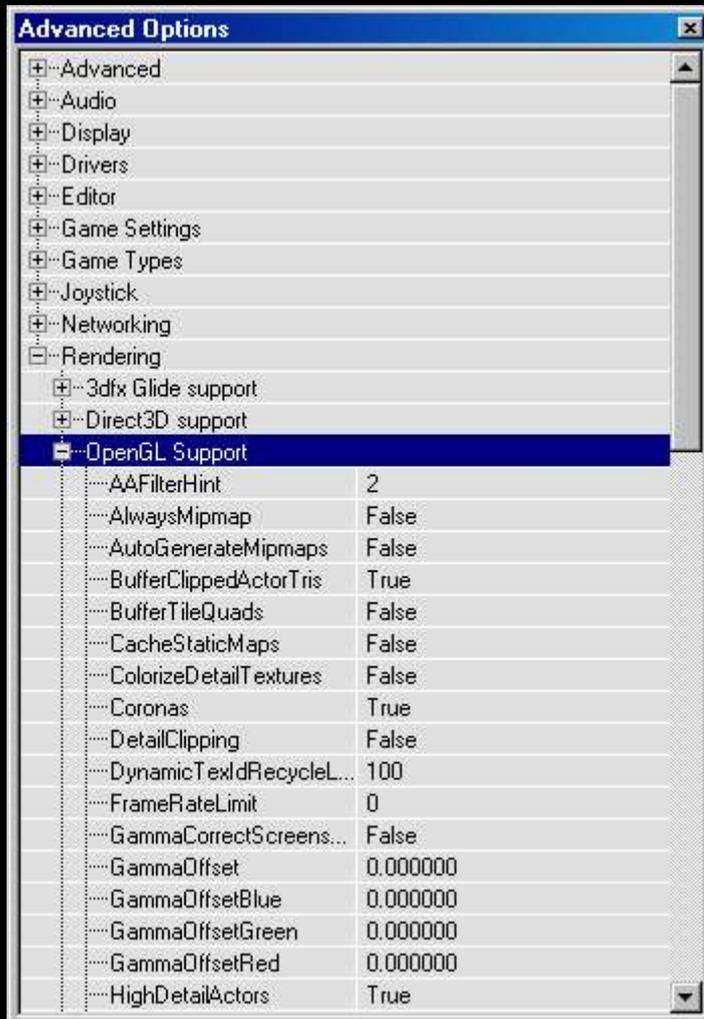
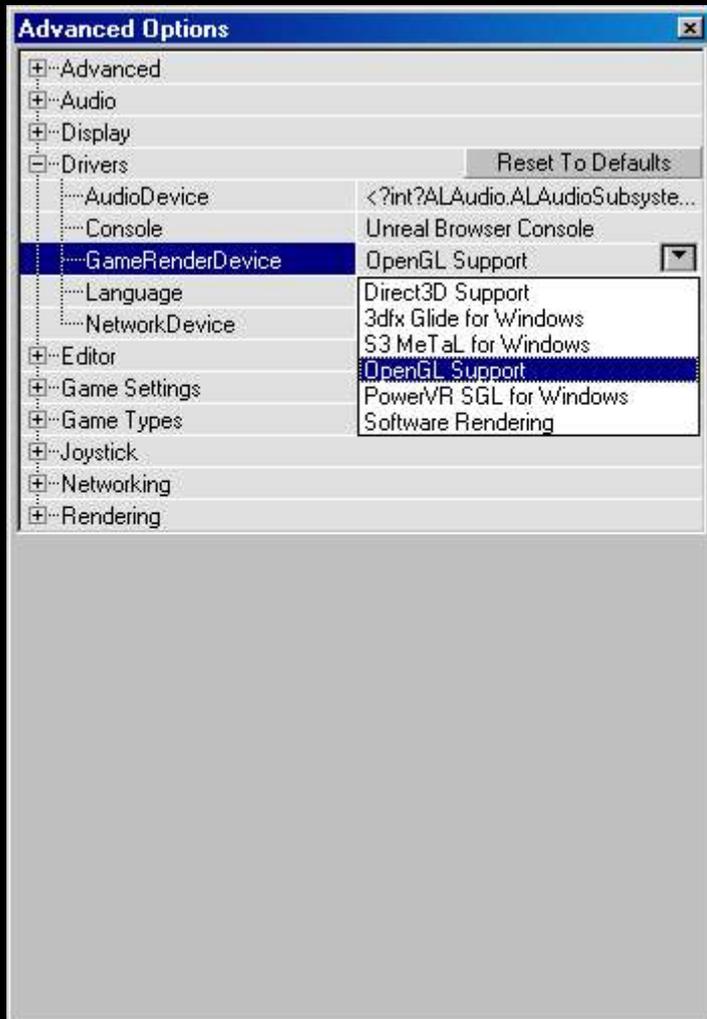
During the install an OpenALwEAX.exe package will be installed as well.(included in the package, the original and new versions can be found at www.openal.org – but if you use a newer version than the one within this package, it may does not work.

USERS OF NVIDIA SOUND CARDS (often onboard) SKIP THE OPENAL INSTALLER AND UPDATE TO THE LATEST NVIDIA SOUND DRIVER (4.57 seems to work), this driver must be loaded separate and is NOT the one included in the driverpackage for the nForce-Chipset.

Be sure to have the latest drivers for your Soundcard installed. If you experience problems try to rename wrap_oal.dll in your \windows\system32 folder (\winnt\system32 for win2k) to openal32.dll (rename the original version to openal32.old or backup it)

To activate the installed files you need to make the following settings in your advanced Options:

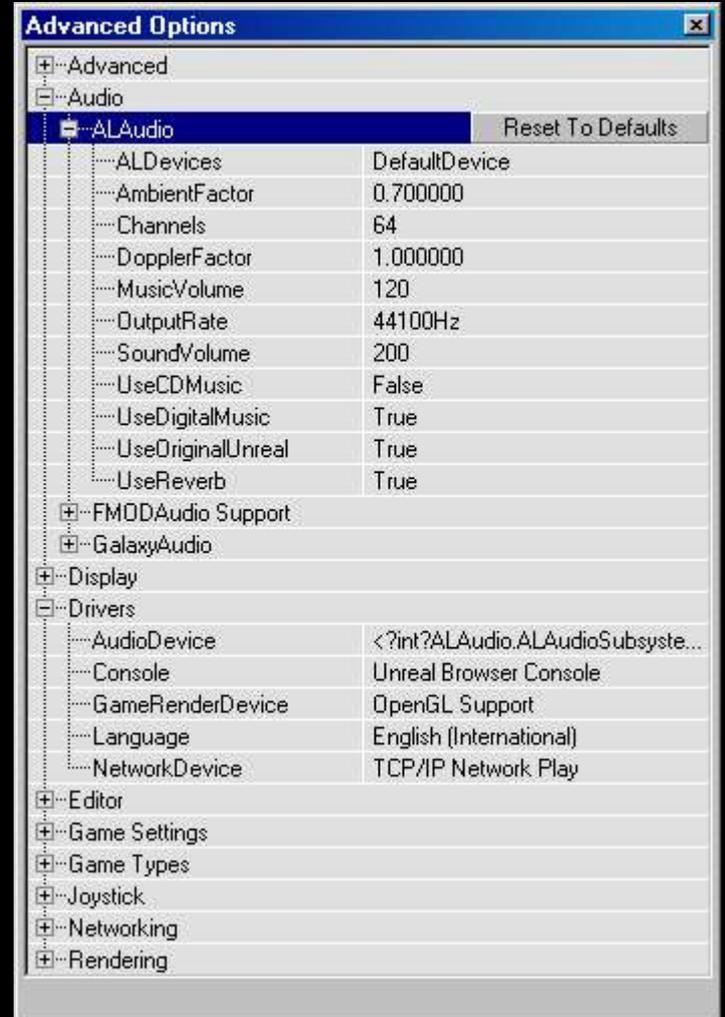
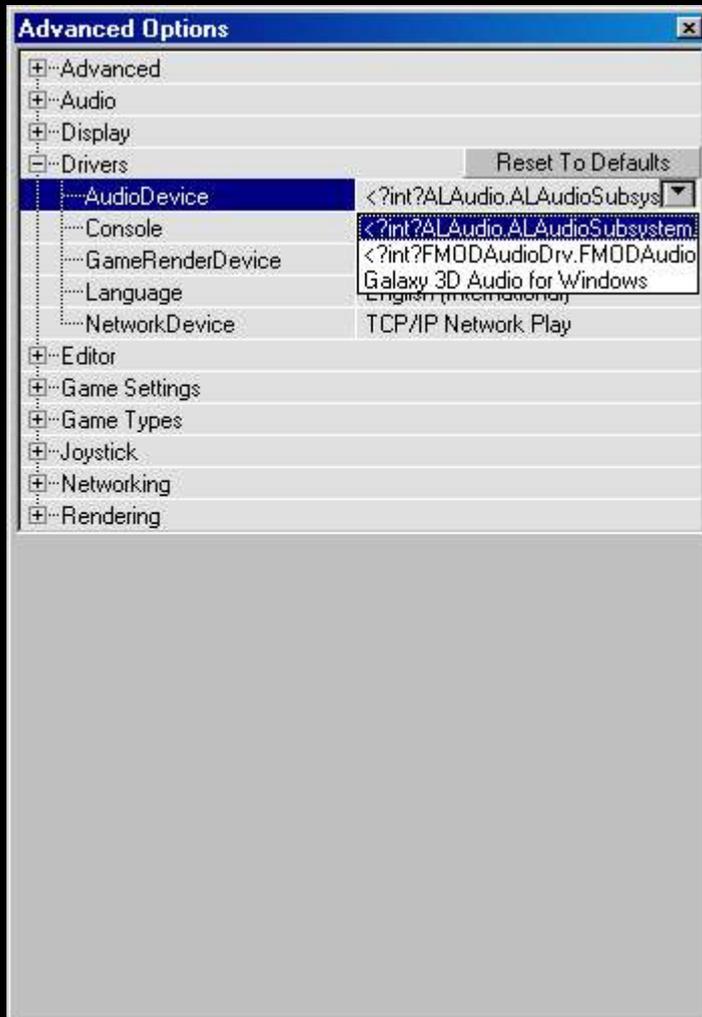
OpenGL in Unreal -> Drivers -> GameRenderDevice -> OpenGL Support and for detailed settings Rendering -> OpenGL Support



in UnrealTournament : Choose inside UT OpenGL in the Video Tab -> Video Driver

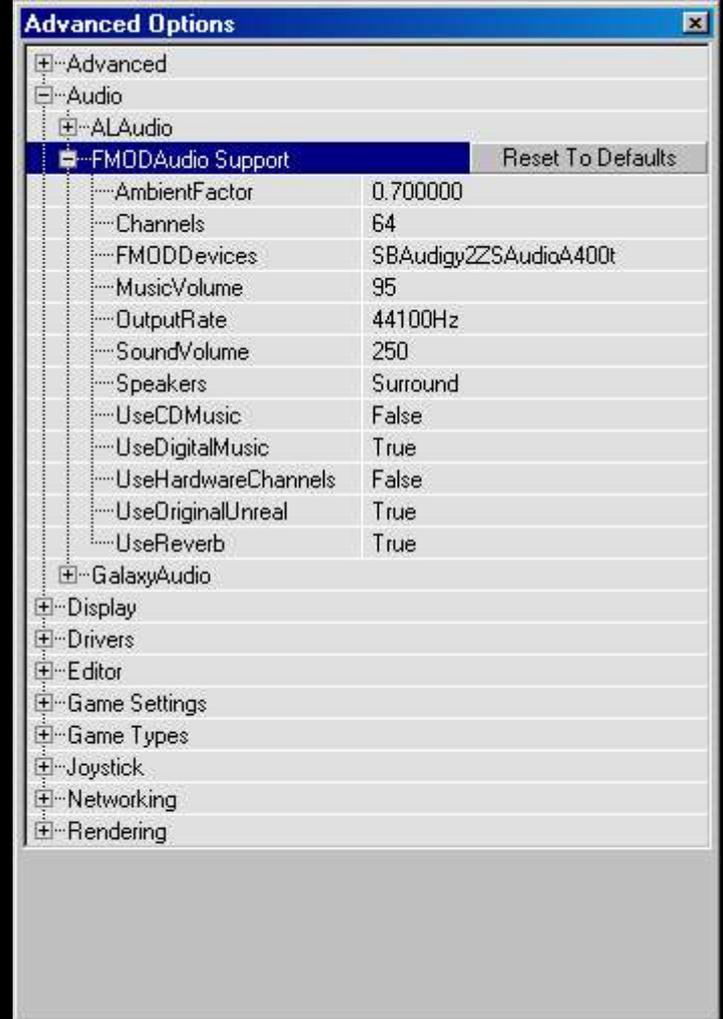
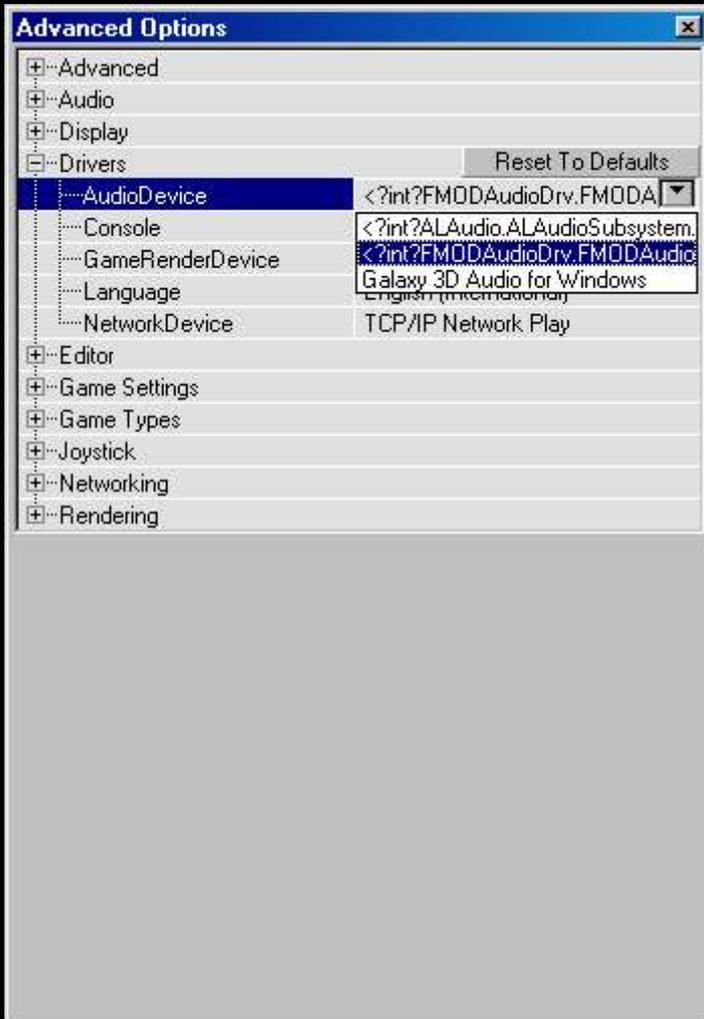
OpenAL:

Drivers -> AudioDevice -> ALAudio.ALAudioSubsystem and for detailed settings Audio -> ALAudio



FMOD:

Drivers -> AudioDevice -> FMODAudioDrv.FMODAudioDevice and for detailed settings Audio -> FMODAudio Support



OpenGL

The OpenGL updates are made to make Unreal work with new grafix-cards, supporting new features and increase speed. One of the advantages is for example to make so called „S3TC“ (High Res, High Color) textures usable with almost every new card. Those textures can be found on the 2nd UT CD (check the tools folder in your Unreal\System dir for the converting tool and a howto) and a new project is working on a reworked textureset for Unreal as well. For more informations visit <http://www.oldunreal.com/cgi-bin/yabb/YaBB.pl?board=S3TC>.

These files are provided by www.oldunreal.com, now with the enhanced renderer code from <http://cwohmal.home.mindspring.com/utglr>

For informations and help visit <http://www.oldunreal.com/cgi-bin/yabb/YaBB.pl?board=2>.

Here are the details for the settings:

UseTrilinear - [True/False]

Controls the use of trilinear texture filtering.

AlwaysMipmap - [True/False]

Can make the renderer always generates mipmaps for textures that are not supplied with them. But, it's always set to 0 by the initialization code (has been this way for a long time), so changing the value of this setting should make no difference.

AutoGenerateMipmaps - [True/False]

Enables the use of the GL_SGIS_generate_mipmap extension for automatic mipmap generation. It is recommended that this setting be disabled as there are far too many video drivers that have unstable, slow, and/or broken support for this extension.

NoFiltering - [True/False]

Can disable filtering on all textures. Useful as a debug option.

MaxAnisotropy - [Integer]

Controls the use and level of anisotropic texture filtering. Disabled if set to 0. Should make no difference if set to 1 (isotropic texture filtering). If set to greater than 1, specifies the maximum degree of anisotropy to use for texture filtering.

UseS3TC - [True/False]

Enables the use of high resolution S3TC compressed textures if they are installed.

Use16BitTextures - [True/False]

Selects lower quality and more compact formats for a number of textures, which will often speed things up. In many cases, there is only minor quality loss. In other cases, like with various skyboxes and coronas, there is often major quality loss.

UseBGRATextures - [True/False]

Allows textures to be uploaded in BGRA format rather than RGBA format if the GL_EXT_bgra extension is supported. This can improve texture upload performance. This option should always be enabled unless it causes problems.

LODBias - [Floating point]

Allows mipmap selection bias to be adjusted. Use negative values to pseudo sharpen textures. Use positive values to blur textures and potentially improve performance at the expense of blurry textures.

UseTNT - [True/False]

A workaround for buggy TNT/TNT2 drivers. Alters texture scaling and mipmap generation behavior. If you really want to know all the details, check the source code.

TexDXT1ToDXT3 - [True/False]

A workaround for poor image quality on NVIDIA GeForce1 - GeForce4 series hardware when using DXT1 format S3TC compressed textures. If enabled, converts all DXT1 textures to DXT3 textures on upload. This improves image quality on the previously mentioned NVIDIA hardware at the expense of twice as much texture memory usage for these textures. The NVIDIA DXT1 image quality problems are most noticeable on certain skybox textures. Keep this in mind when deciding whether or not to trade image quality for speed here. This option should not be enabled on any hardware that draws DXT1 textures with the same quality as DXT3 textures of course.

UseMultiTexture - [True/False]

Controls the use of multitexturing. Should always be enabled as the renderer has a few glitches when it is not. I might try to

track these down some day. Due to the way some parts of the renderer are still written, it is likely to fail on any system without support for the GL_ARB_multitexture extension anyway.

UsePrecache - [True/False]

Controls texture precaching. Texture precaching may improve performance by initializing internal data structures for a number of world textures and most likely getting them loaded into video memory at level load time. It will also slow level loading down some.

MaxTMUnits - [Integer]

Used to limit the number of texture units used by the renderer. Useful as a debug option. Disabled if set to 0.

UsePalette - [True/False]

Controls the use of paletted textures. If there is hardware support for paletted textures, using them can significantly improve performance.

UseAlphaPalette - [True/False]

A workaround for very old buggy GeForce drivers. If set to False, will not upload masked textures as paletted. If there is hardware support for paletted textures, this option should be set to True unless it causes any problems.

MaskedTextureHack - [True/False]

Enabling this option can prevent rendering problems with masked textures when the same texture is applied to different polygons that do not have the masked attribute set consistently across all of them. Likely examples of masked texture problems are rendering errors with solid colored boxes around railings and trees that can often times be fixed with the flush command. There is some risk to using this option, which is why it's called a hack option. It's likely to be very safe, but not completely safe. Implementing it the completely safe way is a lot of extra work, so it uses the simple solution. If it does happen to fail, there will be some completely incorrect textures on some objects.

GammaOffset - [Floating point]

Offset for gamma correction. Can be used to adjust gamma correction even more if you hit the end of the Brightness slider in Video options. The default value of 0.0 causes no change. Use negative values for darker or positive values for brighter. If adjusting this setting for the first time, I'd recommend starting with small values such as -0.3 or 0.3.

GammaCorrectScreenshots - [True/False]

If enabled, will apply gamma correction to screen shots.

GammaOffsetRed - [Floating point]

GammaOffsetGreen - [Floating point]

GammaOffsetBlue - [Floating point]

Fine tuning parameters for gamma correction. These allow different offsets to be specified for each color channel. These offsets are never applied when gamma correcting screen shots, even if GammaCorrectScreenshots is enabled.

OneXBlending - [True/False]

If enabled, matches what the D3D renderer does for blending in multitexture mode when applying lightmaps to world geometry. I can't say for sure which way is correct. In single texture mode, the D3D renderer does appear to do blending like the OpenGL renderer in single texture mode or multitexture mode without OneXBlending enabled.

RequestHighResolutionZ - [True/False]

Allows a high resolution Z buffer to be requested when running in a 16-bit color mode. It's a good idea to enable this option if running in 16-bit color because rendering problems can occur if a 16-bit Z buffer is used. Note that not all video cards support Z and color buffers of dissimilar bit depths.

RefreshRate - [Integer]

Can be used to request a specific refresh rate when running full screen. If set to 0, a default refresh rate is used. If this value is set to an invalid or unsupported refresh rate based on video card or monitor capabilities, the renderer will fail to initialize.

SwapInterval - [Integer]

Controls V Sync. If set to the default value of -1, the default buffer swapping method is used. Set to 0 to disable V Sync.

Set to 1 to enable V Sync. Set to higher values for one frame every N screen refreshes. Not all video drivers support values higher than 1.

FrameRateLimit - [Integer]

CPU controlled frame rate limiter in frames per second. Set to 0 to disable.

UseAA - [True/False]

Enables multisample antialiasing. Requires the GL_ARB_multisample extension.

NumAASamples - [Integer]

Specifies the number of samples to use per fragment for antialiasing. 2 and 4 are common values that should work on many video cards.

AAFilterHint - [Integer]

Can be used to enable Quincunx AA on NVIDIA video cards that support it. Set to 2 to enable this mode.

UseZTrick - [True/False]

Can avoid some z-buffer clears at the expense of cutting z-buffer precision in half. This may improve performance on some video cards. On video cards with z-buffer optimization hardware, enabling this setting may significantly reduce performance as it interferes with some hardware z-buffer optimization implementations.

MaxLogUOverV - [Integer]

Set to 8.

MaxLogVOverU - [Integer]

Set to 8.

MinLogTextureSize - [Integer]

Set to 0.

MaxLogTextureSize - [Integer]

Set to 8, or 0.

UseCVA - [True/False]

Enables the use of the compiled vertex array (CVA) extension. It may be useful on video cards without HW T&L. It is likely to slow things down a little bit on video cards with HW T&L.

UseMultidrawArrays - [True/False]

Enables the use of the GL_EXT_multi_draw_arrays extension.

BufferClippedActorTris - [True/False]

Alters how certain actor polygons are handled, some of which happen to be clipped by higher level code. It's a tradeoff and it is unlikely to make much of a difference either way.

UseSSE - [True/False]

Will auto detect CPU and OS support for SSE instructions and use it if present. Set to False to disable the use of SSE instructions.

UseVertexProgram - [True/False]

Enables vertex program mode. Consider this an experimental option. It can improve performance in some cases. It can also slow things down a lot if certain other settings are not configured correctly. It is likely to slow things down a lot if detail textures are enabled, but single pass detail texture mode is not enabled. It may not work correctly or may cause crashes with some video drivers.

UseTexIdPool - [True/False]

Should be set to True.

UseTexPool - [True/False]

Should be set to True.

DynamicTexIdRecycleLevel - [Integer]

Should be set to the default value of 100.

DetailTextures - [True/False]

Enables detail textures.

UseDetailAlpha - [True/False]

Must be enabled for proper detail texture support.

DetailClipping - [True/False]

Enables the use of a somewhat experimental detail texture mode. It costs more CPU time, but may improve performance in fill rate limited situations.

SinglePassDetail - [True/False]

Enables single pass detail texture mode. This should generally be the highest performance detail texture mode. It requires 4 texture units. It also requires the UseDetailAlpha option to be enabled.

ColorizeDetailTextures - [True/False]

Debug option for detail textures. If enabled, adds a green tint to detail textures

FIX FOR MASKING BUG (by Hellscrag, www.unrealsp.org)

Fixing the black tree bug is easy - it can be done using the console commands.

Save a text file mask.txt to your Unreal\System folder containing the following commands:

```
set tree style STY_Masked
set bird1 style STY_Masked
set krall style STY_Masked
set manta style STY_Masked
set devilfish style STY_Masked
set warlord style STY_Masked
set mantacarcass style STY_Masked
set krallcarcass style STY_Masked
set devilfishcarcass style STY_Masked
set warlordcarcass style STY_Masked
flush
```

Then bind a key in Unreal (I chose M) to:

```
exec mask.txt
```

Revision History :

2.1.0.0

First Release

2.1.0.1

Fixed Detailed Textures Support

2.1.0.2

Added support for adjustable details (number of detail textures applied) for detailtextures

2.1.0.3 small improvements

2.1.0.4 Blackscreen bugfix

2.1.0.5 Release for Unreal Gold and Unreal 226f

2.1.0.6 should fix problems with Radeon cards and resets desktop and gamma after a crash

2.1.0.7 Unreal 226 only

2.1.0.8 First use of the new renderer code by Chris Dohnal

2.1.0.9 Updated to Codesource utglr2.9

Report bugs and comments in the Oldunreal forums <http://www.oldunreal.com/cgi-bin/yabb/YaBB.pl> or mailto

Smirftsch@oldunreal.com with the topic „BUGS OpenGL“ (using anything else it might result in beeing deleted by me due massive spam)

OpenAL

OpenAL is far from being that popular like OpenGL is, the idea is the same, but instead of Video its for Audio. Currently developed and maintained by www.creativelabs.com

A new version of OpenAL will come out soon, and expect this driver to be updated as well then.

What is it for ?

For the same reason i started to make the OpenGL patches i never was satisfied with the original Unreal sound- but since we had to do much work from scratch it took far longer than OpenGL did. OpenAL main target is to support fully 5.1 (Surround) Soundsystems. Best support is on CreativeLabs Cards and probably other High-End Soundsystems. Low-End and Onboard-Audio should do as well, but need more testing. Should do better even on 2 speaker setup than the original Galaxy-Sounddriver.

Set in Advanced Options / Drivers / AudioDevice / ALAudio.ALAudioSubsystem

The advantages of it are:

True Surround Sound (means you now can really hear where your opponents are, 100% precise)

Hardware Sound (how many channels usable depends on the used Soundcard)

Support for Reverb (unfortunately not the original implemented modell, not possible with hardware sound, and maybe never will).

EAX2.0 for different ambients: Zones within Unreal now have a special „ambient“ for every type like Water, Lava, Slime, Nitrogen and Tarzone.

Music Output based on FMOD: www.fmod.org

Here are the details for the settings:

DopplerFactor – [1.0/...]

Factor for doppler effect. If you don't know whats doppler is, leave as is.

UseOriginalUnreal – [True/False]

True : use linear rolloff sound model, like unreal did

False : use inverse distance rolloff model like current game do, (maybe not the best in this case)

UseReverb – [True/False]

True: enable Reverb and EAX effects, including Ambients

AmbientFactor - [0.7/...]

Loudness of unreal ambient sounds, like background sounds (frogs and such), EAX Ambients are not affected by this.

ALDevices

A list of all available Sounddevices in the System. Use DirectSound for Software Rendering, DirectSound3D or specific Creative-Labs Cards (like SB-Live, Audigy) for Hardware

Channels – [64/...]

How many channels used. Depends on Soundcard when used DirectSound3D (Hardware Accelerated) and more or less on CPU-Power and Memory when using DirectSound (Software)

32 Should do it, but sometimes seems to cause crackling. If you experience such problems try to use DirectSound instead with a higher setting. Only most recent soundcards like Creative's Audigy support 64 Hardwarechannels. 64 is recommended.

OutputRate – [8000,11025,16000,22050,32000,44100,48000]

Frequency of the output. Best to leave it at 44100Hz

UseDigitalMusic – [True/False]

Enables/Disables Unreal Music

UseCDMusic – [True/False]

Play Music from CD- never used this myself.

New maps can be enhanced with the following ambients (placed as zoneinfo „EAXZone“ in UED) :

ENVIRONMENT_GENERIC,
ENVIRONMENT_PADDEDCELL,
ENVIRONMENT_ROOM,
ENVIRONMENT_BATHROOM,
ENVIRONMENT_LIVINGROOM,
ENVIRONMENT_STONEROOM,
ENVIRONMENT_AUDITORIUM,
ENVIRONMENT_CONCERTHALL,
ENVIRONMENT_CAVE,
ENVIRONMENT_ARENA,
ENVIRONMENT_HANGAR,
ENVIRONMENT_CARPETEDHALLWAY,
ENVIRONMENT_HALLWAY,
ENVIRONMENT_STONECORRIDOR,
ENVIRONMENT_ALLEY,
ENVIRONMENT_FOREST,
ENVIRONMENT_CITY,
ENVIRONMENT_MOUNTAINS,
ENVIRONMENT_QUARRY,
ENVIRONMENT_PLAIN,
ENVIRONMENT_PARKINGLOT,
ENVIRONMENT_SEWERPIPE,
ENVIRONMENT_UNDERWATER,
ENVIRONMENT_DRUGGED,
ENVIRONMENT_DIZZY,
ENVIRONMENT_PSYCHOTIC

and can be combined (but must'nt be) with specific custom settings:

EnvironmentSize;
Environment Diffusion;
Room;
RoomHF;
DecayTime;
DecayHFRatio;
Reflections;
ReflectionsDelay;
Reverb;
ReverbDelay;
RoomRolloffFactor;
AirAbsorptionHF;

EnvironmentSize:

The Environment Size property sets the apparent size of the surrounding “room.” The value of Environment Size can be considered a characteristic dimension of the room expressed in meters. Scaling Environment Size is equivalent to scaling all dimensions of the room by the same factor.

Because Environment Size is a high-level property, when you change it, it applies a relative adjustment to five lower-level listener properties that determine the shape of the reverberation response: Reflections, Reflections Delay, Reverb, Reverb Delay, and Decay Time. When you change the value of Environment Size, it actually changes the values of the lower-level properties. You can call Get on any of those lower-level properties to see how they have been changed.

The Environmental Audio Extensions 51

Although Environment Size by default affects all five of the lower-level properties mentioned above, you can disable its effect on some or all of these properties by using the Flags listener property described later. It contains one flag for each of the lower-level protocols controlled by Environment Size. All flags are set to TRUE by default to provide a convincing simulation of change in the dimensions of a room (maintaining the reflective properties of its walls). You can override Environment Size’s automatic scaling of any of the lower-level protocols by changing its flag to FALSE. You’ll find more information about this in the description of the Flags property.

Note that when Environment Size is set to a small value (about two meters or less), it adds a coloration effect characteristic of small rooms to the reverberation. This effect becomes stronger as Environment Size is reduced.

Value range: 1.0 to 100.0

Default value: 7.5

Value units Linear meters

EnvironmentDiffusion:

The Environment Diffusion property controls the echo density in the reverberation decay. It’s set by default to 1.0, which provides the highest density. Reducing diffusion gives the reverberation a more “grainy” character that is especially noticeable with percussive sound sources. If you set a diffusion value of 0.0, the later reverberation sounds like a succession of distinct echoes.

Value range: 0.0 to 1.0

Default value: Varies depending on the environment

Value units A linear multiplier value

Room:

The Room property is the master volume control for the reflected sound (both early reflections and reverberation) that EAX adds to all sound sources. It sets the maximum amount of reflections and reverberation added to the sound mix in the primary buffer (the listener). The value of the Room property ranges from 0 dB (the maximum amount) to -100 dB (no reflected sound at all).

Value range: -10000 to 0

Default value: Varies depending on the environment

Value units Hundredths of a dB

RoomHF:

The Room HF property further tweaks reflected sound by attenuating it at high frequencies. It controls a low-pass filter that applies globally to the reflected sound of all sound sources. The value of the Room HF property ranges from 0 dB (no filter) to -100 dB (virtually no reflected sound).

Although the amount and quality of reflected sound controlled by Room and Room HF is global for all sound sources, you can vary the reflected sound for individual sources by setting each source's corresponding sound-source properties—also called Room and Room HF. These source properties treat the listener Room and Room HF properties as a baseline, and are added to the baseline value to determine the final amount of reflected sound for each sound source. (For more information see the description of Room and Room HF in the section on soundsource properties.)

Value range -10000 to 0

Default value Varies depending on the environment

Value units Hundredths of a dB

DecayTime

The Decay Time property sets the reverberation decay time. It ranges from 0.1 (typically a small room with very dead surfaces) to 20.0 (typically a large room with very live surfaces). This low-level property may be controlled by the highlevel listener property Environment Size, in which case its value is scaled according to the value set there. You can disable that automatic scaling by setting the appropriate flag in the listener property Flags. (See its description for more information.)

Value range: 0.1 to 20.0

Default value: Varies depending on the environment

Value units Seconds

DecayHF

The Decay HF Ratio property sets the spectral quality of the Decay Time parameter. It is the ratio of high-frequency decay time relative to the time set by Decay Time. The Decay HF Ratio value 1.0 is neutral: the decay time is equal for all frequencies. As Decay HF Ratio increases above 1.0, the high-frequency decay time increases so it's longer than the decay time at low frequencies. You hear a more brilliant reverberation with a longer decay at high frequencies. As the Decay HF Ratio value decreases below 1.0, the high-frequency decay time decreases so it's shorter than the decay time of the low frequencies. You hear a more natural reverberation.

Value range: 0.1 to 20.0

Default value Varies depending on the environment

Value units A linear multiplier value

Reflections

The Reflections property controls the overall amount of initial reflections relative to the Room property. (The Room property sets the overall amount of reflected sound: both initial reflections and later reverberation.) The value of Reflections ranges from a maximum of 10 dB to a minimum of -100 dB (no initial reflections at all), and is corrected by the value of the Room property. The Reflections property does not affect the subsequent reverberation decay.

You can increase the amount of initial reflections to simulate a more narrow space or closer walls, especially effective if you associate the initial reflections increase with a reduction in reflections delays by lowering the value of the Reflection Delay property. To simulate open or semi-open environments, you can maintain the amount of early reflections while reducing the value of the Reverb property, which controls later reflections.

Value range: -10000 to 1000

Default value: Varies depending on the environment

Value units Hundredths of a dB

ReflectionsDelay

The Reflections Delay property is the amount of delay between the arrival time of the direct path from the source to the first reflection from the source. It ranges from 0 to 300 milliseconds. You can reduce or increase Reflections Delay to simulate closer or more distant reflective surfaces—and therefore control the perceived size of the room.

Value range: 0.0 to 0.3

Default value Varies depending on the environment

Value units Seconds

Reverb

The Reverb property controls the overall amount of later reverberation relative to the Room property. (The Room property sets the overall amount of both initial reflections and later reverberation.) The value of Reverb ranges from a maximum of 20 dB to a minimum of -100 dB (no late reverberation at all).

Note that Reverb and Decay Time are independent properties: If you adjust Decay Time without changing Reverb, the total intensity (the averaged square of the amplitude) of the late reverberation remains constant.

Value range: -10000 to 2000

Default value Varies depending on the environment

Value units Hundredths of a dB

ReverbDelay

The Reverb Delay property defines the begin time of the late reverberation relative to the time of the initial reflection (the first of the early reflections). It ranges from 0 to 100 milliseconds. Reducing or increasing Reverb Delay is useful for simulating a smaller or larger room.

Value range: 0.0 to 0.1

Default value Varies depending on the environment

Value units Seconds

RoomRolloffFactor

The Room Rolloff property is one of two methods available in EAX to attenuate the reflected sound (containing both reflections and reverberation) according to source-listener distance. It's defined the same way as DirectSound's Rolloff Factor, but operates on reflected sound instead of direct-path sound. Setting the Room Rolloff Factor value to 1.0 specifies that the reflected sound will decay by 6 dB every time the distance doubles. Any value other than 1.0 is equivalent to a scaling factor applied to the quantity specified by $((\text{Source listener distance}) - (\text{Minimum Distance}))$. Minimum distance is a DirectSound sound-source parameter that specifies the inner border for distance rolloff effects: if the source comes closer to the listener than the minimum distance, the direct-path sound isn't increased as the source comes closer to the listener, and neither is the reflected sound.

Value range 0.0 to 10.0

Default value 0.0

Value units A linear multiplier value

Air Absorption HF

The Air Absorption HF property controls the distance-dependent attenuation at high frequencies caused by the propagation medium. It applies to both the direct path and reflected sound. You can use Air Absorption HF to simulate sound transmission through foggy air, dry air, smoky atmosphere, and so on. The default value is -0.05 dB per meter, which roughly corresponds to typical condition of atmospheric humidity, temperature, and so on. Lowering the value simulates a more absorbent medium (more humidity in the air, for example); raising the value simulates a less absorbent medium (dry desert air, for example).

Value range: -100.0 to 0.0

Default value: -5.0

Value units Hundredths of a dB per meter

Report bugs and comments in the Oldunreal forums <http://www.oldunreal.com/cgi-bin/yabb/YaBB.pl> or mailto Smirftsch@oldunreal.com with the topic „BUGS OpenAL“ (using anything else it might result in beeing deleted by me due massive spam)

FMOD

Because of its development state this driver is currently not having that many features like OpenAL does, but it supports the original Unreal Reverb (echoes)- like it should be – good examples are Vortex2 and Nyleve. Maybe one day the successor of OpenAL for Unreal, its currently a good choice for low-end cards and 2 Speaker systems. Because fmod seems to be developed itself faster like OpenAL it will maybe replace OpenAL here in Unreal, because of its enhanced features.

Set in Advanced Options / Drivers / AudioDevice / FMODAudioDrv.FMODAudioDevice

Fmods runtime dlls are included in the package, the original and new versions can be found at www.fmod.org – but if you use a newer version than the one within this package, it may does not work.

Be sure to have the latest drivers for your Soundcard installed.

Here are the details for the settings:

UseReverb – [True/False] :

In conjunction with UseHardwareChannels=False this reproduces the original unreal echoes

In conjunction with UseHardwareChannels=True this will play echoes like OpenAL (not yet implemented)

UseOriginalUnreal [True/False]:

True : use use linear rolloff sound model, like unreal did

False : use inverse distance rolloff model like current game do, (maybe not the best in this case)

UseHardwareChannels [True/False]:

True : make use of hardware acceleration on the sound card (you can't have original echoes)

False : use software buffers to process sound (perfect for stereo users)

Speakers [Dolbydigital/Headphones/Mono/Quad/Stereo/Surround]

Your speaker setup

Channels – [64/...]

64 is recommended. Only most recent soundcards like Creative's Audigy support 64 hardwarechannels, so be carefull when chosing UseHardwareChannels. Depends on CPU-Power and Memory when using Software.

AmbientFactor - [0.7/...]

Loudness of unreal ambient sounds, like background sounds (frogs and such), EAX Ambients are not affected by this.

OutputRate – [8000,11025,16000,22050,32000,44100,48000]

Frequency of the output. Best to leave it at 44100Hz

Settings in UED (REVERB in ZoneInfo)

bRayTraceReverb	bool	True – raytrace the reverb effects within this zone
bReverbZone	bool	True – Within this zone reverb is applied to all sounds
CutoffHz	int	Reverb effects are cut off of at this frequency
Delay	int [6]	Time in milliseconds that each reverb stage is delayed
Gain	int [6]	Volume for each of the reverb stages
Master Gain	int	Gain for all reverb effect
SpeedOfSound	float	Speed of sound within zone.

Report bugs and comments in the Oldunreal forums <http://www.oldunreal.com/cgi-bin/yabb/YaBB.pl> or mailto kerilk@oldunreal.com with the topic „BUGS FMOD“ (using anything else it might result in beeing deleted by me due massive spam)