Oldunreal Patch Version 227

If you like those updates you can visit 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.

<u>**This package was created with knowledge and permission of Epic MegaGames, Inc.**</u>

Initial 227 features list (more features and Details in the changelogs below):

A native Linux port is now availible, which runs with OpenGL and SDLSoftDriver for grafix and FMOD for sound.
Built in SuSE, works on LFS as well, but needs glibc2.3.

* New renderers have been added:
OpenGL, D3D8 and D3D9 are completely reworked and heavily improved with the files from UTGLR:
http://cwdohnal.home.mindspring.com/utglr/
These renderers support S3TC Textures and make Unreal compatible with most recent grafix cards.

* New Sounddrivers OpenAL and FMOD:
OpenAL:
Provides hardware support for recent soundcards (best for creative cards), EFX (EAX) reverb soundeffects and full
support for multispeaker setups.
FMOD:
Provides hardware support for recent soundcards, original Unreal reverb soundeffects and full support full support for
multispeaker setups.

* For ban and kickban there are the following commands:
uhelp: Prints the explanations below
uplayers: shows for all players the name, ID, IP-Address, IdentNr and Identity
ukickid: kicks a player with a given ID
ubanid: kicks and bans a player with a given ID (full ban by IP and Name, even after a restart of the game/server)
ubanlist: shows a list with all banned players
uunban: unbans a player with the number X (see in banlist for the ban-number)
utempbanid: kicks and bans a player until server is restarted
utempbanlist: current list of temp-banned players
utempunban: unbans a tempbanned player with the number X (see in tempbanlist for the ban-number)
utempunbanall: unbans all tempbanned players

* Supports footstepsounds and footprints
but to use these new features, the texture properties need to be changed, there are now 4 slots for different sounds
(FootStepSound) the texture produces when walking on, and one variable (Footprint) which defines if, or if not, the
texture shows a footprint when walking on.

* UI-FX added:
Particle Emitters, Weather Simulator, Vegetation Generator and many things more. Details can be found in the forums.

* Supports skeletal meshes like in UT

* Supports HTTP-redirect for downloading maps (like in UT)

* Widescreen adjustment added:
Widescreen users can now adjust the FOV to adjust Unreal better for their screens.
(Maybe some mods will override this setting)

* Added a new setting in NetDrv:
AllowOlderClients True/False
If set to True, 224,225 and UGold Clients can join the server, if False, they will get the Upgrade Message. But either
way, there is a problem. Since the upgrade URL is defined in UpgradeMenu.uc, old clients will be redirected to the old
dead upgrade page, and there is nothing I can do about that.

* Added Coop options for easy configuration of Weapon, Items, Flares and Seeds spawn time
bInstantWeaponRespawn
bInstantItemRespawn
bHighFlareAndSeedRespawn
FlareAndSeedRespawnTime

* A new and fully working quadshot has been added
replaces the old unusable and unfinished quadshot

* Decals have been added to all Unreal weapons (including the monster weapons):
Decals can be added to custom mods as well, but can be turned off if wanted
All weapon decals{or subclasses of Scorch} will have a configurable variable called: DecalLifeSpan.

This variable effects ONLY clients. Servers need not apply.
Default is -1.
-1 = default behavior, the effects will disappear upon not being rendered for some amount of time.
0 = infinite. Decals will NEVER disappear. use at your own risk.
>0 = time in seconds, 1.1 is 1.1 seconds. 30 is 30 seconds.
this is located under "Decal" then "Lifespan" in preferences

* Supports new difficulty levels (up to 6) for Coop & Singleplay:
Needs to be set with command-line parameter ?difficulty=X (X=4,5 or 6)

* Added a new parameter for commandline: -timestamplog
which will force Unreal to put a Timestamp after the logfile name
so if you would simply startup unreal with -timestamplog, the logfile will be:
UnrealYear_Month_Day_Hour_Minute_Second.log
if you start it up with -log=server.log -timestamplog, the logfile will be:
ServerYear_Month_Day_Hour_Minute_Second.log

* New Blood effects:
Gibs now have blood impact decals
If enabled on client and the server is 227c, dynamic blood splatter will occur on all standard(and most mods)
pawns/carcasses
If enabled on server, bleeding will occur, damage is optional.
Blood pools will spawn for killed pawns.

* Added 2 new engine consolecommands:
GameInfo.ConsoleCommand "GetPreLoginAddress", can be used during Event Prelogin call for retreiving the
connectioning client's IP
PlayerPawn.ConsoleCommand "UGetIP", smilar to GetPing but returns instead the Client IP.

* Zoneinfos
If a new damage type and zoneDamageString is supplied in a damaging zone, players will now recieve that death
message.

* Updated UBrowser Server browser:
Bigger default vertical size of that window to show full info of all servers (servername, players, ping, IP, port etc...),
without having to resize it.
Added server version number to appear on server list like in UT.
Added a "Join with password" button option when you right click at a server from the list.
"Join with password" window has an ability to save the password so next time you normally connect to that server
(through UBrowser) it automatly uses that password.
When connecting to an "unknown" server (one where you havent desired any password for) it will join with a randomly
generated password to protect you from password stealing.

* Added a "U227GameRules" class:
can enhance serverside mutator;s, such as modifing player spawn point, modifing damage, preventing deaths,
blocking/modifing chat messages etc...
But note that some of these functions may *not* work with some custom game types that modifies the game.
Also using this class will force your mod to be for 227 use only, so try to keep it in use only on server (a non-
downloadable package) so older clients can still join the server aswell.

* Added a new JumpPad class, UJumpPad

* Added a new setting for Mousehandling to enhance precission with high-resolution mice:
Mouse smoothing can be turned off.

* Now 2 menus are availible:
Unreal classic-style and new UMenu (Unreal Gold) style.

* Naliplayer can now be played ingame

* New implementation allows to use localized chars such as "ö,ä,ü" or other language specific symbols while using "say"

* uses UED2 for Map-Editing

Changelog and additions:

[30.03.08] 227c release

* quadshot accessed nones - fixed

* Player "fell out of the world!" when walking over decorations - fixed

* SoftwareRendering didnt work fullscreen - fixed

* coop server item respawn bug - fixed

* Cant find ByteProperty'VoicePitch' - fixed

* Vortex2 Bug (Playerstart) - fixed

* Sound pitch changes - fixed

* ExtremeLab tubeportals (warpzones) are to slow - fixed

* OldWeapons.u didnt make it into 227b although promised - fixed (sorry for that one)

* The widescreen support is not being applied online - fixed

* UMenu HUD no preview - fixed

* Error handler not working - fixed.

* UMenu - Playersetup - Teamcolor wasn't read out of user settings - fixed

* error not being able to see PreLogin error on Mid-Screen (just as server is at capacity) - fixed

* Fixed player spidering movement issues.

* Fixed player spidering animations to better one.

* Fixed NaliPlayer to use UT'99 NaliPlayer model (one which has weapon triangle).

* Added EngineSubVer variable to LevelInfo to define version number (1=a, 2=b, 3=c).

* Changed ServerQuery to add EngineSubVer for GameVersion number (227 + a/b/c/d etc..).

* Added that older 227 clients (227a, 227b) can't join newer 227 version servers (227c).

* Removed "SpawnPlayActor" and "Incoming travelling actor" logging.

* UBrowser select input keys not working in linux - fixed

* Console set to ";" key in linux

* "shift" key not working by default in linux (ini setting) - fixed

* fixed possible coopgame server crash when monster is attacking a player while leaving.

* umenu player class selection mesh scale - adjusted to work for naliplayer as well.

* OpenAL: some sounds not playing correctly - fixed.

* some weird warpzones crashing Unreal when playing online - fixed.

* fixed bug with skaarj not behaving correctly in vortex2

* another few server fixes for reported crashes (details of these wouldn't help anyone here :))

* added / fixed SDLSoftDrv for Linux - Now there are OpenGL and Software rendering for Linux version.

* fixed SDLGLDrv for Linux. Although maybe a bad idea to use it for playing (it lacks many features and is really not very far advanced), it may be useful for debugging or for some low-end systems. To use it, you need to start it with "./UnrealLinux.bin -noforcesdldrv"

* Enhanced the UWindow EditBox to make it possible to select text with mouse and Copy/Paste text in it.

* Added a Music Player Menu (now default key to open it on F8), which lets you listen to custom musics while playing. Be aware that many anti-cheat mods may not like the custom musics!

* Fixed "Never switch on pickup" option, so client can chose whatever they automatly switch to best weapon on pickup.

* On UMenu.UnrealConsole, you can enable UWindow debug mode (Tools > Debug mode) to debug your own menus while making them.

* Fixed UWindow Win95 LookAndFeel some rendering errors (with menu backgrounds and buttons).

* UMenu: Added Win 95 "LookAndFeel" as selectable GUI interface option now.

* TraverseForm crash (on some maps) - fixed

* fixed some problems with older clients and mods

[05.04.08] 227d release

* Slithprojectile not vanishing on dedicated servers - fixed

* Spawning many Skaarj Player Bots may crash Unreal - fixed

* Translator message for hints - added

* Server wandering port fixed - essential fix for server admin

* DmRetrospective not showing gibs - fixed

* Inventory HUD bug, charge being drawn from one side to another - fixed

* Quadshot:
Pickup sound
Switch to other weapon if 0 ammo
Not being able to select/deselect when 0 ammo but still ammo loaded in the barrels - fixed and size adjusted

* Changed console behavior to display messages without calling it once first

* Black screen bug caused by division by 0 fixed, an obscure bug, will help a few mods.

* Client not directly connecting after downloading a map – fixed

[19.07.08] 227e release

* Fixed OpenAL in Linux version

* Fixed loading savegames in Linux version crashed game

* Fixed UBrowser Menu (classic) for Linux

* Fixed Galaxy reverb crashes.

* Fixed bleeding damage ignoring settings

* Fixed slow download speed for server without redirect. Added a new option: AllowFastDownload:
True = Speed like older servers
False = Reduced speed like pre 227e servers for low bandwidth connection Servers

* Built-in ban manager has been changed to save all bans in a new file called "Security.ini"

* Fixed so last jump dosent cause an "ghost" jump in online games.

* Fixed support with multiple jumpboots (custom mods).

* Fixed broken shield effect on online games.

* Fixed UED2 crashed when using MyLevel

* Fixed UED2 crashed when closing it

* Fixed UED2 group browser checkbox was inverted

* Fixed UED2 after you load script with longer name, with first letter of script with name you are trying to load you CANT load it (for example, you cant loading Male after loading MaleOne)

* Added support for playing special Nali/Male 1/Female 2 skins.

* Added support for importing and playing Ogg Vorbis music files (in editor on Music Browser or in game as map music). (For OpenAL and FMod use only!)

FMod:
Added in ALAudio alike "AStat Audio" and "AStat Detail" commands.
Fixed the reveb issue with maps like QueenEnd.
Added in a new option to enable Sound Attenuate (sounds that are played behind walls are heard in lower volume).

* Added IP/Name/ID playerlogger for Servers and messaging it for server admins/server log which will stored in Security.ini too if GameInfo bMessageAdminsAliases/bLogNewPlayerAliases are true. Like the banning system, this feature is only available for 227 clients.

* Added Web-Administration support
A WebServer administration page.
Features:
-Restart map - Restart currently playing map.
-Switch map - Switch map to selected map/game/mutators (note that maplists configure must be enabled to access maplist).
-Current game - You see list of players (ID/Name/Ping/Score/IP) and controls to Kick/Ban them.
-Server console - See a list of chat messages in server aswell a command line where you can chat with the players to execute a command on server.
-Banlist - See a full list of all banned clients, also lets you unban them.
-Defaults - Server configure page:
* Main Game config - Lets you configure basic game rules (such as max players/server packages/server actors/redirecting etc).
* ServerInfo config - Lets you configure public server rules (such as server name/admin name/MOTD).

* Maplists - lets you configure maplists (for DeathMatch based games, NOT for Coop).
* Mod Configures - Custom mods configures page (by default it contains configures for DeathMatch, TeamGame, Cooperative).
-Changed to little better interface and added a config option for the new 227e "AllowFastDownload" flag.
Read the included UWebReadme.txt for more info.

* Added in an "OverridePrelogin" function for GameRules to modify prelogin allowance or something (its called directly after GameInfo Prelogin).

* Added in a new class 'PlayerClassManager' which allows custom mods to temporarly adding in to player settings some new player classes/skins for that game only.

* Actor:
Added a new variable "bNetNotify" and a new event "PostNetReceive", can be used by mod authors for notifing whenever a replicated variable has been changed on client (whenever bNetNotify is True).

* UBrowser:
Added in a config option to set initial startup page (Advanced Options > Networking > UBrowser > InitialPage).

* Added some new native functions to enhance UScripting:
Object:
native(1700) final function Object FindObject(Class ObjClass, string ObjectName);
-Find an object based on object name/class.
native(1701) final function Class GetParentClass(Class ObjClass);
-Get Parent class of a desired class.
native(1702) final iterator function AllObjects(class BaseClass, out Object Obj);
-Iterate through all objects in game.
native(1723) final iterator function AllFiles(string FileExtension, string FilePrefix, out string FileName);
-Iterate through all Unreal Package files (u, umx, utx, uax...).

Actor: native(1714) final function Actor SpawnAct(Class ActClass, vector Loc, optional rotator Rota, optional name ActName, optional Actor Own, optional Pawn Instigat, optional Actor Template, optional bool bMayColFail);
-Spawn an actor class with some additional parameters. native(1722) final function bool CanReachPoint(vector Start, vector End, float ColRadius, float ColHeight, float JumpZ, float XYSpeed);
-A point reachability function for AI use.
static native(1713) final function NativeExec(string Cmd);
-Pretty much same as 'ConsoleCommand' except it's static function.

LevelInfo:
-static native(1703) final function byte GetConState(NetConnection Other);
-Get connection state out of a net connection.
static native(1704) final function string GetConIP(NetConnection Other, out int Port);
-Get net connection IP aswell as Port.
static native(1705) final function string GetConOpts(NetConnection Other);
-Get net connection options (?Name=Bla?Class=Bla?etc...).
native(1706) final function bool HasDownloaders();
-Fast check if current game has some downloaders.
native(1707) final iterator function AllConnections(out NetConnection Connect);
-Iterate through all connections currently on server.
native(1708) final iterator function AllDownloaders(out NetConnection Connect, out string File, out int Sent, out int TotalSz);
-Iterate through all downloaders on server.
native(1712) final function PointRegion GetLocZone(vector Pos);
-Get the zone out of a desired location.
native(1720) final function Object AllocateObj(Class ObjClass);
native(1721) final function FreeObject(Object Obj);
-For storing temporarly objects/actors that can be reused later.

NavigationPoint:
-native(1709) final function int GenReachSpec(Actor Start, Actor End, int Dist, int ColR, int ColH, int RchFlgs, bool bPruned);

-Generate reachspec for current map.
native(1710) final function bool EditReach(int Idx, optional Actor Start, optional Actor End, optional int Dist, optional int ColR, optional int ColH, optional int RchFlgs, optional bool bPruned);
-Edit an excisting reachspec in current map.
native(1711) final function bool RemoveReachSpec(int Idx);
-Remove a reachspec from the current map.

Canvas:
native(1715) final function Draw2DLine(Color Col, vector Start, vector End);
-Draw a 2D line on the screen.
native(1716) final function Draw3DLine(Color Col, vector Start, vector End);
-Draw a 3D line in world.
native(1717) final function vector WorldToScreen(vector WorldPos);
-Convert world coordinates to screen coordinates.
native(1718) final function vector ScreenToWorld(vector ScreenPos);
-Convert screen coordinates to world coordinates.
native(1719) final function DrawPathNetwork(bool bOnlyWalkable);
-Render current map's path network (with 3D lines).
native(1724) final function coords GetCameraCoords();
-Get the current camera location and rotation.

* Added a new command for server admins : admin UGetConnections it shows current connection ips, player names, and downloading status
* Added a new emitter particle system (Emitter.u/Emitter.dll) which is handeled 99 % in C++ scripts (giving a huge speed boost), currently it supports:
-Normal sprite emitter
-Mesh emitter
-Weather emitter
- Beam emitter

Full variables explanation:
XEmitter:
EmGeneral:
bDisabled - Emitter is disable, no more spawning new particles.
bRespawnParticles - Respawn particles that have died.
bAutoDestroy - Auto-destroy emitter actor after all particles have died (can be used for temp effects).
bAutoReset - Auto-reset emitter after all particles have died and AutoResetTime has passed.
bSpawnInitParticles - Spawn initial particles, or else wait for possible auto-reset (only when bRespawnParticles is false).
bStasisEmitter - Smilar to Actor bStasis, but stop updating particles once player isnt seeing the emitter zone.
MaxParticles - Maximum amount of particles.
ParticlesPerSec - Amount of particles to spawn per second (0 = auto assign the value).
LifetimeRange - How long time particles should live.

EmRevolution:
bRevolutionEnabled - Particle revolving should be enabled.
RevolutionOffset - Revolving offset for the particles.
RevolutionsPerSec - Revolving speed.

EmVisuals:
ParticleTextures - Random/Animation sprite frames for particles.
bUseRandomTex - Should use random frame or else animate the textures.
ParticleStyle - The style of the particles.
StartingScale - Starting scale of the particles.
TimeScale - Time scaling of the particles (timescale 0-1).

EmFade:
FadeInTime - Fade in time scale (0-1).
FadeOutTime - Fade out start time (0-1).
FadeInMaxAmount - Max scale glow when fully faded in (0-2).

EmPosition:

SpawnPosType - Should use Box/Sphere/Cylinder spawn offset?
BoxLocation - Box spawning offset for particles.
SphereCylinderRange - Sphere/Cylinder offset range.
bRelativeToRotation - Spawning offset should be in relative to actor rotation.
bUseRelativeLocation - All particles location/rotation/velocity should be relative to actor?

EmTrigger:
TriggerAction - Emitter actor triggering action.
SpawnParts - If spawn particles, then how many?

EmSpeed:
SpeedScale - Particles speed scale in relative time (0-1).
ParticleAcceleration - Particles acceleration range.
SpawnVelType - Whatever it should use Box/Sphere/Clinder velocity for the particle.
BoxVelocity - Box velocity for particles.
SphereCylVelocity - Sphere/Clinder range.
bVelRelativeToRotation - Velocity should be relative to actor rotation?
bCylRangeBasedOnPos - Sphere/Cylinder velocity range should be relative to particle spawn offset?

EmCollision:
ParticleCollision - Collision type: Nothing/Walls/All Actors/Projectile target actors.
ParticleExtent - Particle collision size.
ParticleBounchyness - Bounchyness of the particles (when hitting wall/actor).

EmLight:
bLightParticles - Particles should have lighting.
PartLightBrightness - Particle light brightness.
PartHue - Light hue.
PartSaturation - Light saturation.
PartLightRadius - Light radius.

EmSound:
ImpactSound - Particle impact sound (when hitting wall/actor).
SpawnSound - Particle spawn sound.
DestroySound - Particle destroy sound.

XMeshEmitter:
EmMesh:
ParticleMesh - Display mesh of the particle.
bRenderParticles - Should render mesh in "bParticles".
bParticlesRandFrame - Should particles mesh use random texture animation frame.

EmRotation:
ParticleRotation - Particles rotation type.
bRelativeToMoveDir - Particles should have rotation relative to movement direction.
RotationsPerSec - Rotation rate of particles.
InitialRot - Initial rotation of the particles.

EmAnim:
bAnimateParticles - Particles should animate.
ParticleAnim - Particle anim sequence.
PartAnimRate - Particle anim rate.
bPartAnimLoop - Particle should loop animation.

XWeatherEmitter:
PartTextures - Random texture for the particles.
Position - Spawning offset around player camera.
AppearArea - Appearance area around actor location (when AppearAreaType is Area).
Lifetime - Particle lifetime.
Speed - Particle speed.
Size - Particle size.

Force - Particles force.
WeatherType - Whatever the weather type is Rain/Snow/Dust (falling type).
ParticleCount - Maximum number of particles.
PartStyle - Style of particles.

Realtime view support in Editor

Usage:

Chose your prefered renderer during the first run after installation:

OpenGL, Direc3D8 and Direct3D9

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://cwdohnal.home.mindspring.com/utglr
For informations and help visit http://www.oldunreal.com/cgi-bin/yabb/YaBB.pl?board=2.

Here are some 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 was 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 or 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.
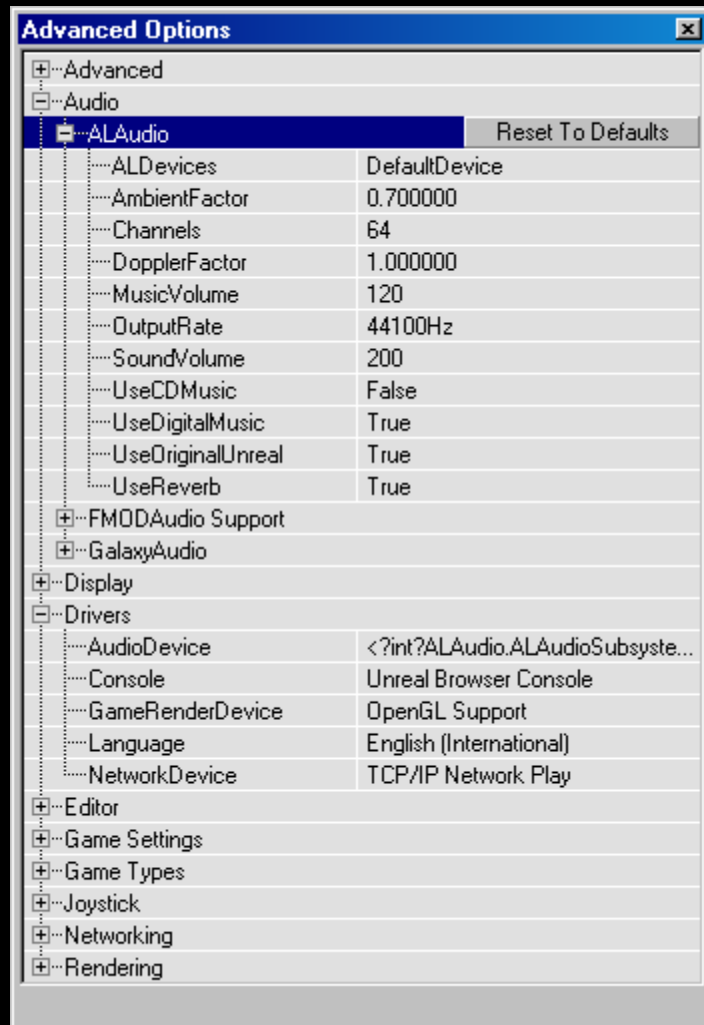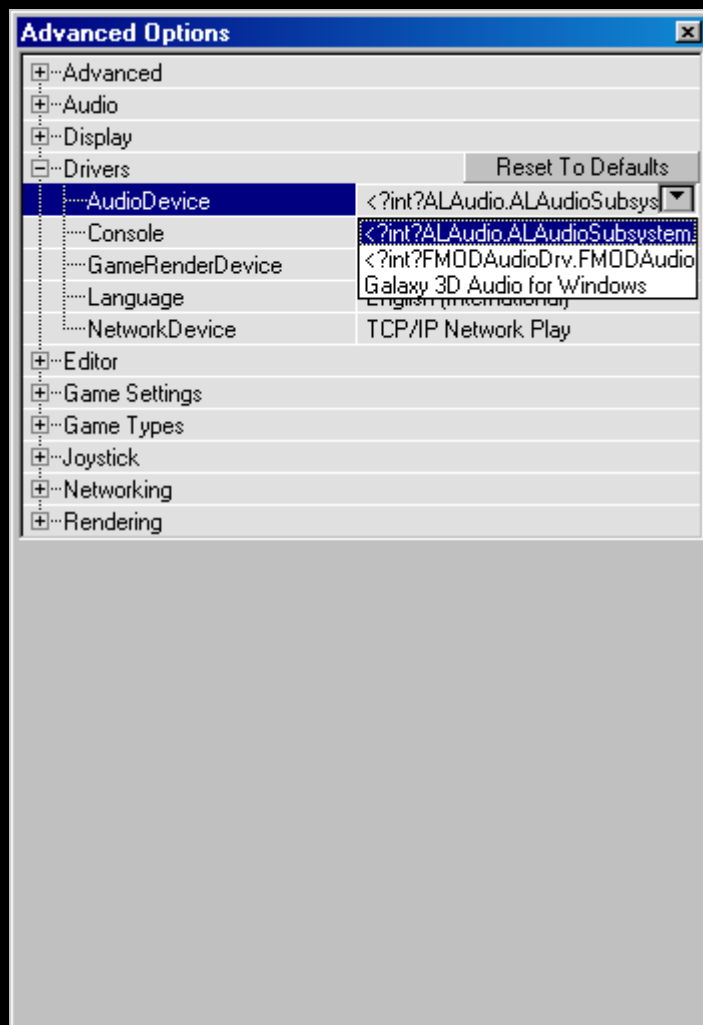
ColorizeDetailTextures - [True/False]
Debug option for detail textures. If enabled, adds a green tint to detail textures

OpenAL

OpenAL:

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

OpenAL is far from beeing 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 orginal 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 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) :

```
    REVERB_PRESET_GENERIC
    REVERB_PRESET_PADDEDCELL
    REVERB_PRESET_ROOM
    REVERB_PRESET_BATHROOM
    REVERB_PRESET_LIVINGROOM
    REVERB_PRESET_STONEROOM
    REVERB_PRESET_AUDITORIUM
    REVERB_PRESET_CONCERTHALL
    REVERB_PRESET_CAVE
    REVERB_PRESET_ARENA
    REVERB_PRESET_HANGAR
    REVERB_PRESET_CARPETTEDHALLWAY
    REVERB_PRESET_HALLWAY
    REVERB_PRESET_STONECORRIDOR=
    REVERB_PRESET_ALLEY
    REVERB_PRESET_FOREST
    REVERB_PRESET_CITY
    REVERB_PRESET_MOUNTAINS
    REVERB_PRESET_QUARRY
    REVERB_PRESET_PLAIN
    REVERB_PRESET_PARKINGLOT
    REVERB_PRESET_SEWERPIPE
    REVERB_PRESET_UNDERWATER
```

```
REVERB_PRESET_DRUGGED
REVERB_PRESET_DIZZY
REVERB_PRESET_PSYCHOTIC
REVERB_PRESET_CASTLE_SMALLROOM
REVERB_PRESET_CASTLE_SHORTPASSAGE
REVERB_PRESET_CASTLE_MEDIUMROOM
REVERB_PRESET_CASTLE_LONGPASSAGE
REVERB_PRESET_CASTLE_LARGEROOM
REVERB_PRESET_CASTLE_HALL
REVERB_PRESET_CASTLE_CUPBOARD
REVERB_PRESET_CASTLE_COURTYARD
REVERB_PRESET_CASTLE_ALCOVE
REVERB_PRESET_FACTORY_ALCOVE
REVERB_PRESET_FACTORY_SHORTPASSAGE
REVERB_PRESET_FACTORY_MEDIUMROOM
REVERB_PRESET_FACTORY_LONGPASSAGE
REVERB_PRESET_FACTORY_LARGEROOM
REVERB_PRESET_FACTORY_HALL
REVERB_PRESET_FACTORY_CUPBOARD
REVERB_PRESET_FACTORY_COURTYARD
REVERB_PRESET_FACTORY_SMALLROOM
REVERB_PRESET_ICEPALACE_ALCOVE
REVERB_PRESET_ICEPALACE_SHORTPASSAGE
REVERB_PRESET_ICEPALACE_MEDIUMROOM
REVERB_PRESET_ICEPALACE_LONGPASSAGE
REVERB_PRESET_ICEPALACE_LARGEROOM
REVERB_PRESET_ICEPALACE_HALL
REVERB_PRESET_ICEPALACE_CUPBOARD
REVERB_PRESET_ICEPALACE_COURTYARD
REVERB_PRESET_ICEPALACE_SMALLROOM
REVERB_PRESET_SPACESTATION_ALCOVE
REVERB_PRESET_SPACESTATION_MEDIUMROOM
REVERB_PRESET_SPACESTATION_SHORTPASSAGE
REVERB_PRESET_SPACESTATION_LONGPASSAGE
REVERB_PRESET_SPACESTATION_LARGEROOM
REVERB_PRESET_SPACESTATION_HALL
REVERB_PRESET_SPACESTATION_CUPBOARD
REVERB_PRESET_SPACESTATION_SMALLROOM
REVERB_PRESET_WOODEN_ALCOVE
REVERB_PRESET_WOODEN_SHORTPASSAGE
REVERB_PRESET_WOODEN_MEDIUMROOM
REVERB_PRESET_WOODEN_LONGPASSAGE
REVERB_PRESET_WOODEN_LARGEROOM
REVERB_PRESET_WOODEN_HALL
REVERB_PRESET_WOODEN_CUPBOARD
REVERB_PRESET_WOODEN_SMALLROOM
REVERB_PRESET_WOODEN_COURTYARD
REVERB_PRESET_SPORT_EMPTYSTADIUM
REVERB_PRESET_SPORT_SQUASHCOURT
REVERB_PRESET_SPORT_SMALLSWIMMINGPOOL
REVERB_PRESET_SPORT_LARGESWIMMINGPOOL
REVERB_PRESET_SPORT_GYMNASIUM
REVERB_PRESET_SPORT_FULLSTADIUM
REVERB_PRESET_SPORT_STADIUMTANNOY
REVERB_PRESET_PREFAB_WORKSHOP
REVERB_PRESET_PREFAB_SCHOOLROOM
REVERB_PRESET_PREFAB_PRACTISEROOM
REVERB_PRESET_PREFAB_OUTHOUSE
REVERB_PRESET_PREFAB_CARAVAN
REVERB_PRESET_DOME_TOMB
REVERB_PRESET_PIPE_SMALL
REVERB_PRESET_DOME_SAINTPAULS
REVERB_PRESET_PIPE_LONGTHIN
REVERB_PRESET_PIPE_LARGE
REVERB_PRESET_PIPE_RESONANT
REVERB_PRESET_OUTDOORS_BACKYARD
REVERB_PRESET_OUTDOORS_ROLLINGPLAINS
REVERB_PRESET_OUTDOORS_DEEPCANYON
REVERB_PRESET_OUTDOORS_CREEK
REVERB_PRESET_OUTDOORS_VALLEY
REVERB_PRESET_MOOD_HEAVEN
REVERB_PRESET_MOOD_HELL
REVERB_PRESET_MOOD_MEMORY
REVERB_PRESET_DRIVING_COMMENTATOR
```

```
REVERB_PRESET_DRIVING_PITGARAGE
REVERB_PRESET_DRIVING_INCAR_RACER
REVERB_PRESET_DRIVING_INCAR_SPORTS
REVERB_PRESET_DRIVING_INCAR_LUXURY
REVERB_PRESET_DRIVING_FULLGRANDSTAND
REVERB_PRESET_DRIVING_EMPTYGRANDSTAND
REVERB_PRESET_DRIVING_TUNNEL
REVERB_PRESET_CITY_STREETS
REVERB_PRESET_CITY_SUBWAY
REVERB_PRESET_CITY_MUSEUM
REVERB_PRESET_CITY_LIBRARY
REVERB_PRESET_CITY_UNDERPASS
REVERB_PRESET_CITY_ABANDONED
REVERB_PRESET_DUSTYROOM
REVERB_PRESET_CHAPEL
REVERB_PRESET_SMALLWATERROOM
```

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

```
EFXflAirAbsorptionGainHF GCC_PACK(4);
EFXflDecayHFRatio;
EFXflDecayLFRatio;
EFXflDecayTime;
EFXflDensity;
EFXflDiffusion;
EFXflEchoDepth;
EFXflEchoTime;
EFXflGain;
EFXflGainHF;
EFXflGainLF;
EFXflHFReference;
EFXflLFReference;
EFXflLateReverbDelay;
EFXflLateReverbGain;
EFXflReflectionsPanX;
EFXflReflectionsPanY;
EFXflReflectionsPanZ;
EFXflRoomRolloffFactor;
```

Details can be found in the EffectsExtensionGuide.pdf file which is included in this package.

FMOD

FMOD:

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

:
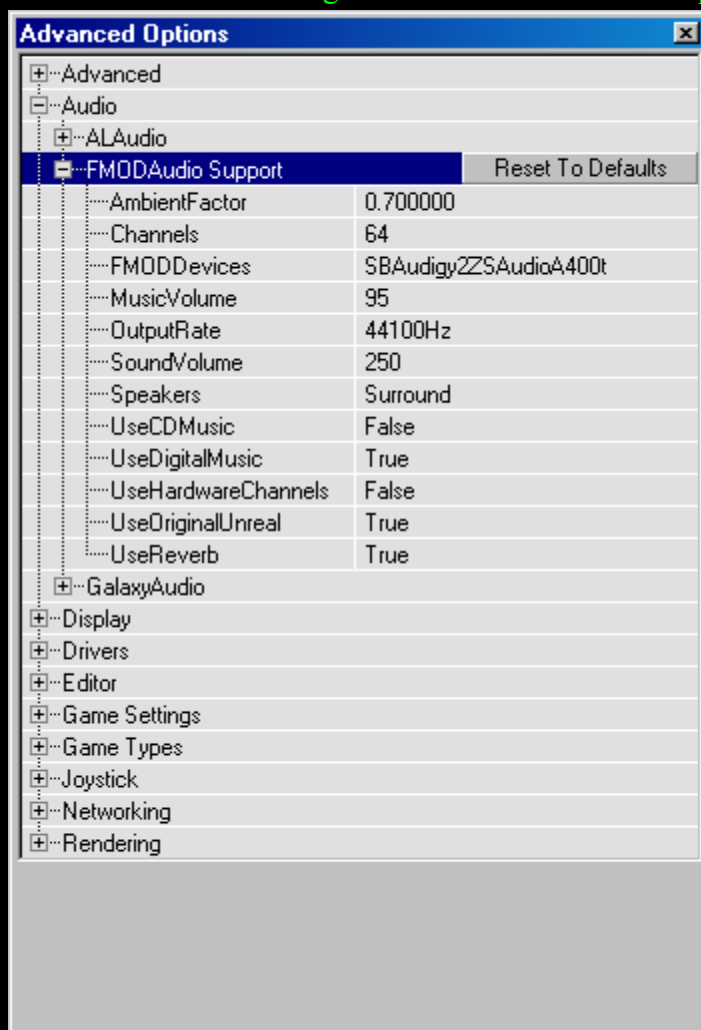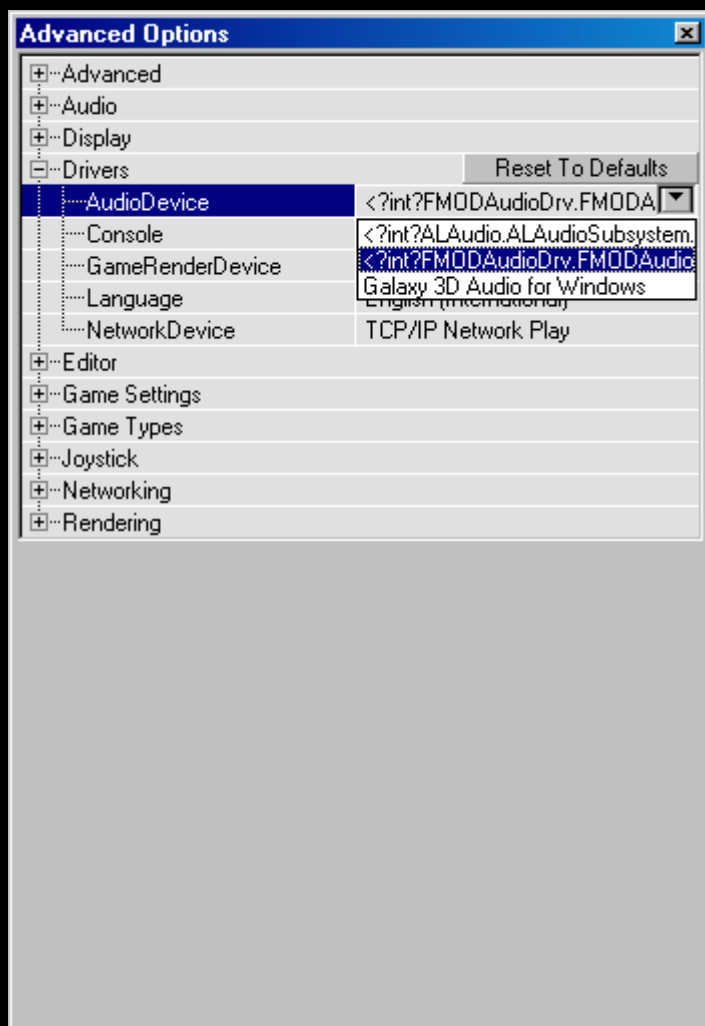
Because of its development state this driver is currently not having that many features like OpenAL does, but it supports the original Unreal Reverb (echoes)- like it should be – good examples are Vortex2 and Nyleve. Maybe one day the successor of OpenAL for Unreal, its currently a good choice for low-end 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

Special Thanks:

Here is a small list with people that helped me in this project (no special order, without any single person of them It would have been much harder if not impossible).
Please forgive me and gimme a notice if I missed someone, but over all the time my tortured brain is maybe failing...

Epic for giving me this chance!

UTPG for their work in UT which helped me very much

Shambler - native coding, anticheat, many things more and being patient with my stupid questions Wink

Wolf - Anticheat and help in Uscript, help in adding features, advices and being there for me and this project anytime

Smpdev for his great OpenGL and D3D8, and advices in native coding.

Kerilk for FMOD and helping me with OpenAL and being patient with my stupid questions also

Asgard12000 for tons of script fixes

Zombie for script-fixes, security issues

[]KAOS[]Casey Script fixes, blood Effects, testing, bug hunting

..:..: Script fixes, native Coding additions (such as Particle-Emitter), bug hunting, many other additions
and many others now for uscript advice and fixes

Bozo, for tons of fixed meshes- maybe not that obvious as other things, but must have been a hell a lot of work, and can be seen if you watch all the lovely details.

Raven for nice addons and new features like ParticleEmitter

Krull for his 227 map DMRetrospective

SA-Digimes for hosting my Oldunreal-Serpentine server, which I needed badly for testing

DieHard for his great High-Resolution Textures, which make 227 more beautiful than ever

HyperNL for his Enhanced ServerBrowser and for his intense server testing

Turboman for his skeletal mesh testing and UMenu background

My girl for helping me with decals and especially for being patient Tongue

Shivaxi for bughunting, some footstepsounds

Pitbull for a nice chat in the night

All friends who offered me mirroring

The community here which helps me to find and fix the bugs

And of course all I may forgot now and all who donated to Oldunreal, helping me to pay and maintain Oldunreal