WARCRAFT® III
Art Tools

Introduction

Thank you for installing the Warcraft III Art Tools. With this set of tools, you can create your own custom models and animations for use with Warcraft III: Reign of Chaos and Warcraft III: The Frozen Throne.

PLEASE NOTE: Blizzard Entertainment does NOT directly support the Warcraft III Art Tools. Our technical support team will not answer questions about art tool functionality and will not be able to help resolve problems you may experience while using the art tools.

System Requirements

You must have a valid license for Discreet's 3ds max 4 in order to use these art tools. We highly recommend that you use 3ds max version 4.26 with the service pack 1 fixes. Windows 2000 or Windows XP is suggested. To make textures, you will need a paint program, such as Adobe's Photoshop.

Tool Descriptions

The art tools include the following plug-ins for 3ds max:

Model Exporter plug-in (ModelExp)
Warcraft III Material plug-in (War3bmtls)
Warcraft III Preview (SGView)
User Property Editor (UserProp)
Warcraft III Particle plug-in (BlizPart)
Warcraft III Ribbon plug-in (Ribbon)

Each plug-in is described briefly below.

Model Exporter Plug-in

You will use this plug-in to export your custom model. This plug-in handles converting the model from 3ds max data into data that Warcraft III can use. It also handles converting and exporting your model textures into the correct format for Warcraft III. See Exporting the Model from 3ds max for more information.

Warcraft III Material Plug-in

A *material* is a combination of render states (such as transparency, textures, and lighting); Warcraft III uses this combination to draw models on the screen. You will use the Blizzard Material plug-in to manage these render states. The Model Exporter also uses the Blizzard Material plug-in to export rendering information to an output file. See Texture Maps and Materials for more information.

Warcraft III Preview

When activated, the Model Previewer opens a 3D window that displays your current model and animations in the same way that Warcraft III will display them. This tool allows you to more rapidly test your model and animations during development. See Using Warcraft III Preview or more information.

User Property Editor

Some information for Warcraft III models is stored in 3ds max user properties. You can use the User Property Editor to change these properties via a 3ds max rollout panel instead of typing them by hand. See The User Property Editor for more information on the User Property Editor and the properties that it can help you change.

Warcraft III Particle Plug-in

The Warcraft III Particle plug-in provides a custom particle system type. You will use this particle system in your models to create particle effects in the game. The plug-in has a large rollout panel with many parameters that alter how the particle effects look in Warcraft III. Many effects are possible by changing these parameters. See Warcraft III Particle System for more information.

Warcraft III Ribbon Plug-in

The Warcraft III Ribbon plug-in provides a custom ribbon object type. You will use this object in your models to create trail-like ribbon effects. The plug-in has a rollout panel that includes many parameters for customizing ribbon appearance and behavior. Many effects are possible by changing these parameters. See Ribbon Helpers for more information.

Glossary of Terms

Attachment Point – *Attachment points* are locations in space where Warcraft III will attach additional art to a model in game. Aura effects and spell effects are examples of graphical effects that depend on attachment points. Although attachments are created using a helper object that appears to have a spatial volume, only the origin of that volume is considered to be the origin of the model. Attachment points are specified in model space: that is, relative to the model's origin. Thus, a model's attachment points move with the model.

Billboard – A *billboard* is a piece of geometry that is procedurally oriented to face the camera. Fully billboarded objects will always face the camera in the same orientation. Objects that are only billboarded on one or more axes will face the camera on those axes, but may rotate around other axes.

Blend Mode – The *blend mode* refers to the technique used to combine multiple pixel layers in order to form a final image. Blend modes include *additive*, which means that pixels' color values are added together (saturating at pure white); *modulate*, which means that pixels' color values are multiplied together; and *blend*, which means that new pixel color and the current frame buffer pixel color are blended together using the new pixel alpha value as a blending factor.

BLP – A *Blp* is an external file containing all the textures used to display some object in the game. Each of these textures is stored in a file ending in the extension .BLP. BLP files are automatically created for you when you export a model.

Depth Buffer – The *depth buffer* is a section of memory on the 3D graphics hardware. This buffer is generally used to keep track of the closest object seen so far at each pixel location. Depth buffering is used to reject pixels or entire triangles that are farther away than the nearest objects that have been drawn so far. This process accomplishes pixel-accurate depth sorting of objects in a scene. Keep in mind that sometimes depth buffer tests or sets are disabled, such as when you are drawing transparency effects.

Draw Order – Warcraft III establishes a *draw order* for all of the models in the scene. This refers to the order in which different parts of a model are drawn. For models, the draw order is as follows: Models are split into opaque and transparent layers. All opaque layers are drawn at the same time. The ordering of opaque layers doesn't matter, because depth buffering ensures that only the per-pixel closest elements are rendered to the frame buffer. Transparent layers are sorted from back to front, and then rendered in that same back-to-front order after all of the opaque layers have been drawn.

Event – In Warcraft III, *events* are triggers embedded in the model that cause footprints to appear; splats, uber-splats, or spawned models to be emitted; or sounds to play.

Frame Buffer – The *frame buffer* is a section of memory on the 3D graphics hardware that accumulates all of the triangles that have been drawn so far in a scene. When everything is drawn, the frame buffer is "swapped" or "presented", and the graphics hardware actually sends the image to the monitor. There are usually two copies of the frame buffer in use: one is being shown on the monitor, while the other one is accumulating information for the next scene. This technique is known as *double-buffering*.

Geometry – This term refers to the set of vertices and faces that make up an object. Conceptual object types such as a sphere must be reduced to real geometry (triangles) before they can be used by the game.

Geosets – This is a Warcraft III concept. Geometry in Warcraft III is divided up into *geosets* based on material and mesh properties. Generally, each combination of textures and material properties (such as unlit or no-depth-test) causes a separate geoset to be generated. All triangles in a particular geoset are rendered at the same time; this approach is more efficient for the rendering hardware. Ideally, you want to generate as few geosets as possible.

Hit Test/Collision – When you select a unit in Warcraft III, the game must perform a computation called a *hit test* in order to determine what you clicked on. Hit testing takes quite a bit of processing, so we use various techniques to make it faster. One of these techniques is to hit test against a simpler shape than the model itself. Two very simple shapes to hit test against are spheres and boxes. Warcraft III uses both of these types of simplifications to make hit testing faster. The Warcraft III Art Tools will generate *collision* shapes for you automatically. These collision shapes are used by the hit testing algorithms.

MDX/MDL – A model is stored on disk in one of two formats: MDX or MDL. Thus, depending on which format was used, a model file will have a filename that ends in either .MDL or .MDX. *MDX* files contain a compact binary representation of the model. The art tools export model files in MDX format by default. *MDL* files contain a textual representation of the model. This format is inefficient for Warcraft III to load and parse, but it's straightforward for debugging. You can choose to output models in MDL format for debugging.

Mip Levels – Modern 3D graphics hardware uses a technique called *mipmapping* to increase image quality when the size of a textured image is much smaller than the size of the source texture. Mipmapping requires that progressively smaller versions of a texture are generated and made available for use by the graphics hardware. Each successively smaller version of a texture is ¼ of the size of the original texture (half the original size along each dimension), clamped to a minimum of 1 pixel wide. Each version is known as a *mip level*, and the collection of mip levels is frequently called a *mip chain*. The Warcraft III Art Tools automatically generate mip levels for textures that are used on models.

Model – A collection of data used by Warcraft III to draw an image on the screen.

Origin – *Origin* refers to the reference point in a system. In the Model Previewer, the origin is at (0,0,0), which is located by default in the center of the screen. In 3ds max, each scene has an origin, and each object has an origin. When you select an object in 3ds max, you will see an axis tripod appear at that object's origin.

Parent/Child – The Warcraft III Art Tools use the same parent-child concept that 3ds max does. When you link two objects together, they form a *parent-child* relationship. The *parent* is the object that is linked TO. The *child* is the object that is linked FROM.

Queue – A *queue* is a first-in-first-out arrangement of data. Think of the line at the bank: the first person in line is the first person to be serviced when a teller becomes available.

Sequence – A *sequence* is a time-slice of animation from the overall animation timeline for a model. A sequence represents one of the model's actions in game. For example, if the object is a unit, the object will probably need walk, stand, attack, birth, and death sequences. Models generally include multiple sequences to represent these states visually.

Team Color – This color is procedurally applied to an object in order to identify which team the object belongs to. See Replaceable Texture ID for more information.

Team Glow – This is a texture that has the same color as team color. Just as team color is applied to an object, team glow is applied to a unit in order to identify the unit's team. Team glow is most often used on Hero units.

Tris – Shorthand for "triangles". This is the collection of interconnections between vertices that define the shape of a 3D object.

Unit – A single playable entity in Warcraft III, such as a Peon or an Archmage. Each unit in Warcraft III is generally represented by a single model, although there are some exceptions.

Verts – Shorthand for "vertices". This is the collection of locations in 3D space that define the points of interconnection for a 3D object.

Building a Warcraft III Model

In this next section we will go over the details of the features used to export a model from 3ds max to Warcraft III. Both Warcraft III: Reign of Chaos and Warcraft III: The Frozen Throne use the same model format and features. We will also give a basic tutorial for making a simple model along the way. A basic knowledge of 3ds max is expected.

3ds max to game scale

In Warcraft III the unit scale in 3ds max is one inch, so a Peon is roughly 70 units tall. The tallest buildings are approximately 300 units tall. A pathing cell is 32 units wide, and a terrain cell is 128 units wide. A single cliff height is exactly 128 units high. The origin in 3ds max is always the origin of the unit, and a unit stands with its center over the origin. Also, the unit faces you while it is in the front view.

Setting up the model geometry

Geometry requirements

There are very few limits on geometry as long as it is made of polygons. Animating vertices and morphs are not supported. How the geometry appears on the first frame of animation (we will call this frame 0) is how it will appear in Warcraft III. Smoothing groups are directly translated to vertex normals per tri.

Simple Unit Tutorial, step 1

Build a sphere with a radius of 50 that rests on the origin and use an Edit Mesh modifier to deform it so that it has an eye or two as well as a mouth. Later we will make the mouth animate open and closed.

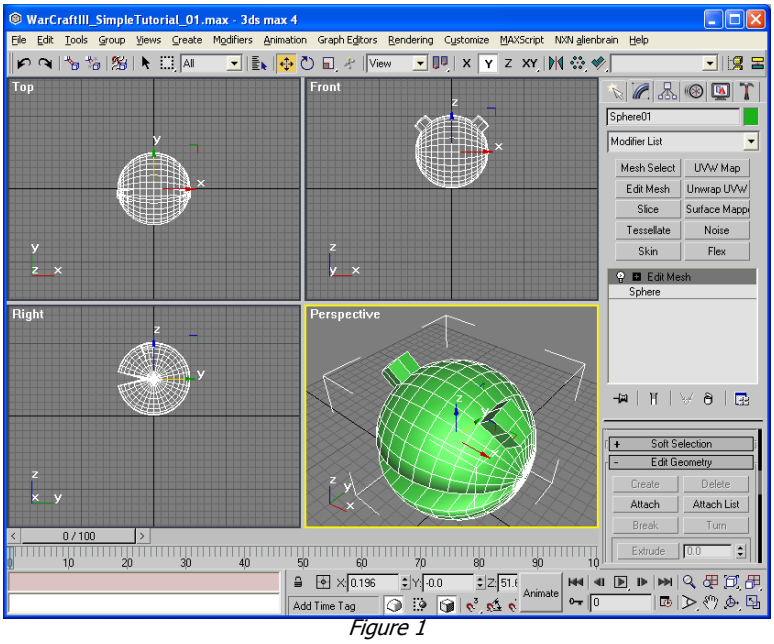

Figure 1

Texture Maps and Materials

Texturing requirements

Textures are assigned in the diffuse color map channel as 24-bit or 32-bit .TGA files. A texture's size must be some power of 2, with a max size of 512x512. Neither the height-to-width ratio nor the width-to-height ratio can be greater than 8 to 1. The alpha channel is where you can draw team color or create transparent areas in the model. White in the alpha channel is opaque, and black is transparent.

Mip levels

The BLP process of exporting textures makes all versions of mip levels needed using a simple bi-linear filter. Sharpened or hand-altered mip levels can be created by saving additional textures with the suffix of "_mip1.tga", "_mip2.tga", etc. Each step of a mip level texture is half the original texture's height and width. These extra mip levels are not assigned in 3ds max, but automatically detected on preview or export when they are in the same directory and named correctly.

Material requirements

The only types supported are Warcraft III and Composite. A composite material allows multiple Warcraft III materials to draw as multiple passes in the game. Multi-object or sub-object materials and geometry with multiple material IDs are not supported.

The Warcraft III Material Plug-in

This plug-in may seem daunting at first for its sheer number of options. These options are listed for advanced reference.

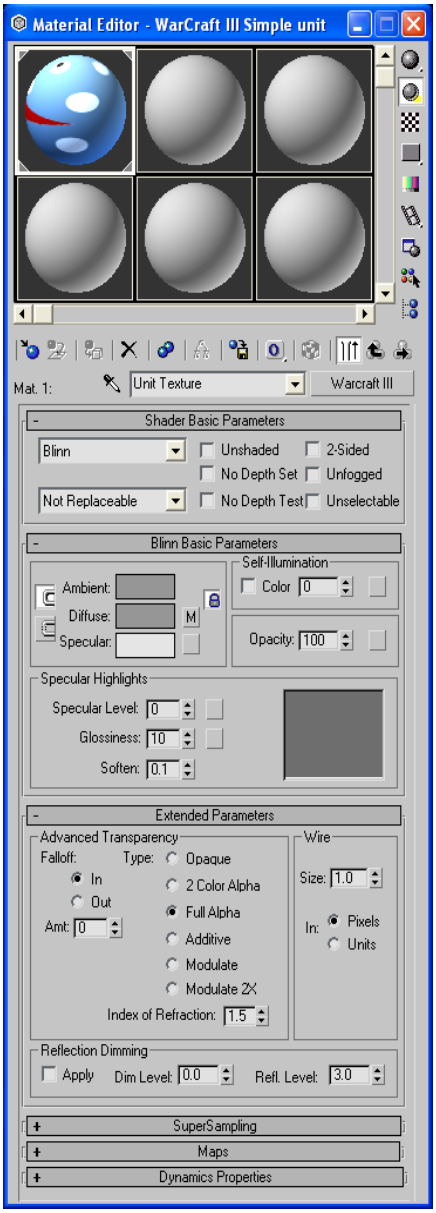

Figure 2

Options in the shader basic parameters

Shader Type – This option must remain as Blinn.
Replaceable Texture - This pull-down menu will allow Warcraft III to procedurally alter the texture. This option is required for team color. Leave this on Not Replaceable almost always.
Unshaded - This option causes the material to be fully lit and ignore light sources.
No Depth Set – The object's polygons will not write to the depth buffer. This option is ignored when using Additive, Alpha, Modulate, and Modulate 2X blend modes.
No Depth Test - The object's polygons will not test the sort order from the depth buffer.
2-Sided – The polygons will always draw and not use backface culling. The lighting is determined by the one normal for the faces, so one side will always shade incorrectly.
Unfogged – An unfogged geoset will ignore darkening from distance fog.
Unselectable – A geoset that is flagged as unselectable is ignored for hit-test collision testing.

Options in Blinn basic parameters

Self-Illumination – The texture and lighting are multiplied by this color.
Opacity – In additive and alpha blend modes, this value directly affects how transparent the geoset is. Furthermore, the opacity can be animated.

All other Blinn basic parameters are ignored.

Options in extended parameters

The only options that are used are under Advanced Transparency if you choose the blend mode for the geoset.
Opaque (default) – The geoset draws as solid, with no blend to the frame buffer.
2 Color Alpha – The geoset draws as solid except for regions over which the texture map alpha channel is black, where the geoset is completely transparent. The cutoff from transparent to solid is 75% white in the alpha channel. This blend mode can have a fixed or animating opacity channel.
Full Alpha – Standard blend mode using an 8-bit alpha channel texture and/or opacity channel. Depth set is turned off. Draw order is in the transparent queue. Overlapping tris can cause errors in sort order. This is the slowest-drawing blend mode.
Additive – The geoset adds its RGB values to the frame buffer, causing a ghost-like glowing effect. Most spells use this to make glowing effects. Depth set is turned off. Draw order is in the transparent queue. No sort order errors with its own geoset.
Modulate – The geoset multiplies its RGB values to the frame buffer, making a semi-transparent darkening of the background. Depth set is turned off. Draw order is in the transparent queue. Opacity has no effect. No sort order errors with its own geoset.
Modulate 2X – The geoset first doubles its initial RGB values, then multiplies them against the frame buffer. This technique causes both brightening and darkening in one pass. Depth set is turned off. Draw order is in the transparent queue. Opacity has no effect. No sort order errors with its own geoset.

Options in the maps panel

Textures are assigned in the diffuse map channel of the Blizzard material. A sequence of .TGA files can also be assigned, creating an .IFL, but you must select Use Path. The amount is ignored. All other maps are ignored.

Options in the bitmap sub-panel – coordinates

The following selections are directly supported: Explicit Map Channel, Map Channel number, Tile check box. All other options are ignored.

Simple Unit Tutorial, step 2

Using a paint program, make a texture to be mapped from the side view of the Simple Unit. Save it as a 256x256 32-bit .TGA file. There should be some large dots in the alpha channel for the team color to show through (see Figure 3 & 4).

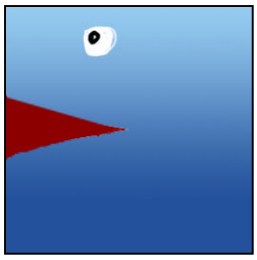

Figure 3

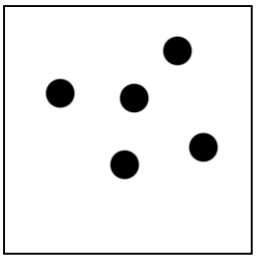

Figure 4

We will be making a two-pass material, with a team color texture for the first pass and the unit map we just made as the second pass. In the 3ds max Material Editor, change the material type from Standard to Composite. Now set the base material to the Warcraft III type. Change the Replaceable pull-down field from Not Replaceable to Team Color. Any map that is now in the diffuse color slot will be ignored, but the tools still need a map to be there. We also suggest that you check the unshaded box for team color on units. That's it for the first layer.

Back to the composite material, now make the Mat. 1 Slot a Warcraft III type. This time assign the .TGA file we made in the diffuse map channel. Lastly this layer needs to be Full Alpha blend mode in the Advanced Transparency section of the Extended Parameters.

Assign this material to our Sphere Unit's geometry.

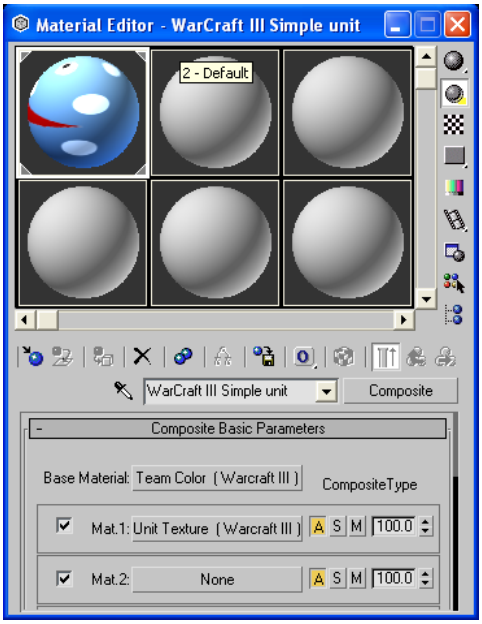

Figure 5

UV Mapping

Supported mapping types

The UVW Map and Unwrap UVW modifiers are supported for one non-animating map channel. UVW Map can also animate when in planar mapping. The Unwrap UVW modifier(s) should be left in the Modifier Stack and not collapsed. XYZ mapping is not supported. Visual errors can occur if two adjacent vertices share the same UV. Multiple UV map channels can be used with planar mapping. Texture tiling is controlled by the material and not by the mapping. Flipping has no effect.

Simple Unit Tutorial, step 3

Add a UVW Map modifier and fit it to the planar map from the side. You may need to rotate the UVW Gizmo to the right angle. You can now add an Unwrap UVW to fine-tune areas like the eyes.

At this point you can preview your work with the Warcraft III Preview. Under the Utilities Panel chose the Warcraft III Preview and click the button Preview. This will launch the viewer to look at your model.

If any errors are detected, they will open in a dialog box showing the warning or a fatal error. Warnings can indicate a problem, but the Previewer will still try to display your work. Fatal errors will prevent the launch of Preview. At this time if something is wrong, go back over the first steps.

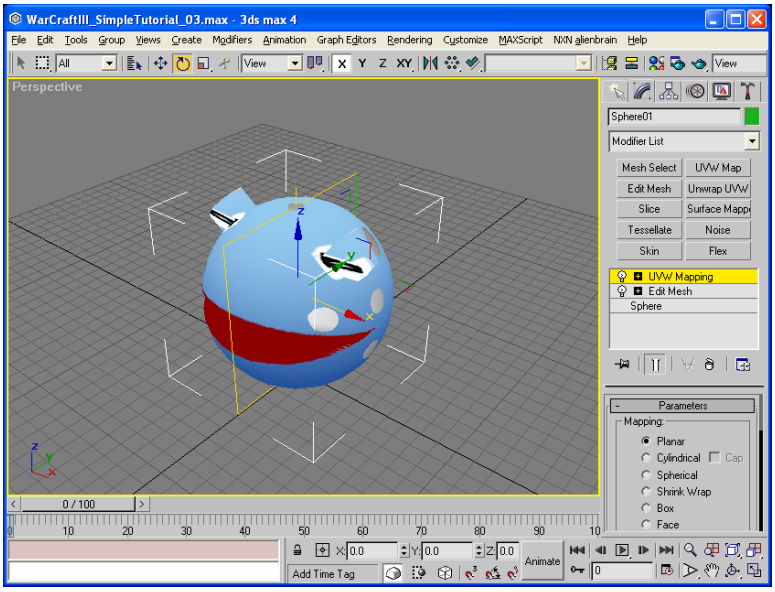

Figure 6

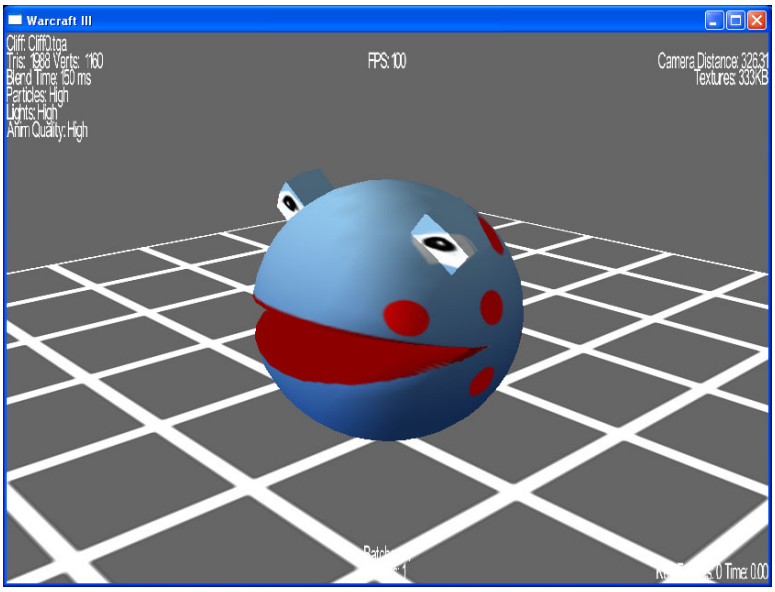

Figure 7

Setting Sequences

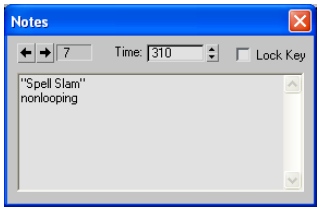

Figure 8

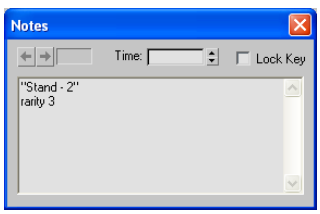

Figure 9

Warcraft III uses "note track" keys in the track view to define animation sequence length and timing. One 3ds max file includes all sequences for a Unit Model. These note track keys also define whether the sequence plays one time or loops. The note track for sequences is always under Objects in the track view. See Animation List of sequence names for more information.

Sequence note track requirements

The sequence name must be identical on the two keys that mark the beginning and end of a sequence. No other note keys can be between them.

Sequence name requirements

Sequence animation names are one or more word tokens separated by spaces. The sequence name must be in quotes if it includes multiple tokens. The full sequence name is made up of primary and secondary name tokens like **"Stand Ready",** and these animations are chosen by the game engine by a best-match method. In the case of a unit that attacks and then pauses between attacks, the game code will ask the unit to play the sequence "Stand Ready". If that animation doesn't exist, the unit will fall back to its stand animation. There are numerous possible combinations of animations, and you can further customize the animation sequence by playing with the timing of those animations.

Note track options

Comment - Often for reference only, a *comment* can be added to the name; this comment is ignored by the game engine. The Warcraft III artists used such comments to name animation versions by simply adding a space and a dash to the name. All text after the "–" and still inside the quotes is ignored.

NonLooping – By default, all animations loop. If an animation is to play exactly one time, the line **NonLooping** can be added. Attack and spell animations are NonLooping, since they play one time and then play the stand (or "Stand Ready") animation before their cooldown allows them to attack or cast a spell again.

MoveSpeed - The comment **movespeed** followed by a space and a number doesn't affect the model at all, but it is a useful tool in previews, where it will move the ground grid at that rate to simulate walking. Any value can be used. Most Warcraft III units have between 250 and 400 movement speed.

Rarity – Warcraft III's animation choosing system allows you to have multiple animations with the same name. The animations are randomly chosen. To ensure that certain animations happen less often, add the line **rarity** followed by a space and a single number from 1 to 9. The higher the number is, the less often that animation will be chosen.

Time Scaled Animations – Certain sequences are given a fixed time so they can be procedurally altered in playback speed. Birth and decay animations are usually set up with a sequence length of exactly 1800 frames, or 1 minute. The animations will be scaled faster or slower by the game engine to equal the build time or decay time specified for that unit.

Simple Unit Tutorial, step 4

Open the track view and add a note track under Objects. The first sequence we need to make is a walk. Add a key to the note track at frame 10 and another at frame 40. At a default time scale of 30 frames a second, this will make a one-second walk animation. Select both of the keys and right-click one. In the Notes pop-up, type:

Walk

MoveSpeed 300

Just like figure 10. This will make a walk sequence.

For this Simple Unit we will also need stand, attack, and death sequences. All of them should be 30 frames in length and have about 20 frames between sequences. Both attack and death will need **NonLooping**. *If you preview now, nothing will change, and the sequence will be ignored until the model actually animates. That's the next part.*

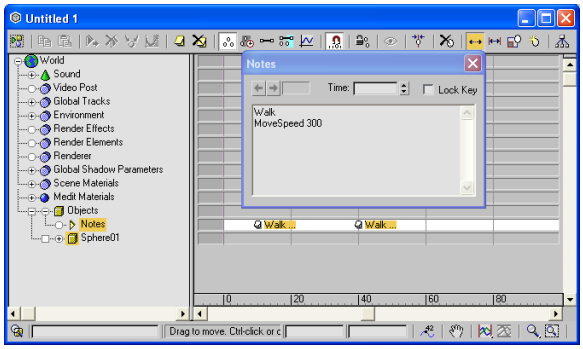

Figure 10

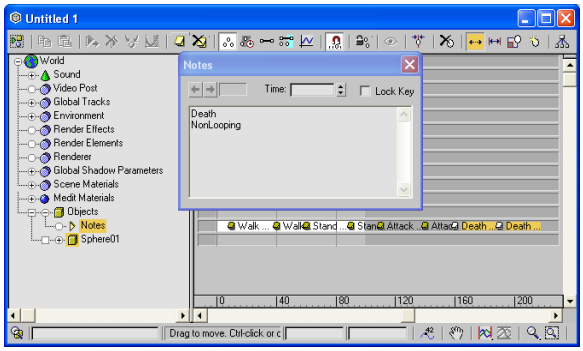

Figure 11

Animating Geometry, Bones and Helpers

Geometry can be directly animated with position, rotation and scale. Animating non-uniform scaling is not the same as it is in 3ds max unless the origin of the scaling object is aligned with the world in frame 0. For complex animations we suggest you make a skeleton and link the geometry to that skeleton.

Acceptable skeleton types

The animating character skeleton can be made of max bones, helpers, and geometry, such as an editable mesh. Geometry needs to have the "bone" user property flagged, or it will actually draw the geometry. These bone types need to be unhidden when you are exporting them. Units should have a bone (or a geometry flagged as a bone) named "bone_head" and another named "bone_chest", both of which should be placed in the skeleton respectively. These are used by the Warcraft III Editor script "Lock body-part facing". Likewise, turreted buildings can use "bone_turret" to aim only the turret at the target.

Animation controller types

Position/Rotation/Scale controllers must be Bezier, Linear or TCB. Opacity and visibility can be Bezier, Linear TCB or On/Off. The Parameter Curve Out-of-Range types can be set to Constant (default) or Cycle (for global looping animation). IK chains or character studio bipeds are not supported.

The Bezier controller can make a no-interpolation stepped sequence by using the Bezier Key Info, In and Out Tangents, Step type. For more information, use 3ds max help and search for "Bezier step tangent type". This approach requires all the keys in that controller track to be of the Tangents to Step type, but it can give instant transforms quite unlike other animation controller blended transforms.

On/Off visibility controllers are treated as status toggle switch and can be inverted by a parent's On/Off controller. If a parent bone's visibility track is animated off and the child's is animated off as well, the model can become corrupted in Warcraft III, because its origin node has been removed from the animation, yet the child becomes visible. This can cause strange effects, like children bones that become attached to random objects.

Key frame requirements

Rotation controllers need to have a key every 90 degrees or less. If there are no keys on the first and last frame of a sequence, the default position, frame 0, will be used. On/Off visibility controllers are the only controller type that doesn't require a key frame on the first and last frame of a sequence.

Simple Unit Tutorial, step 5

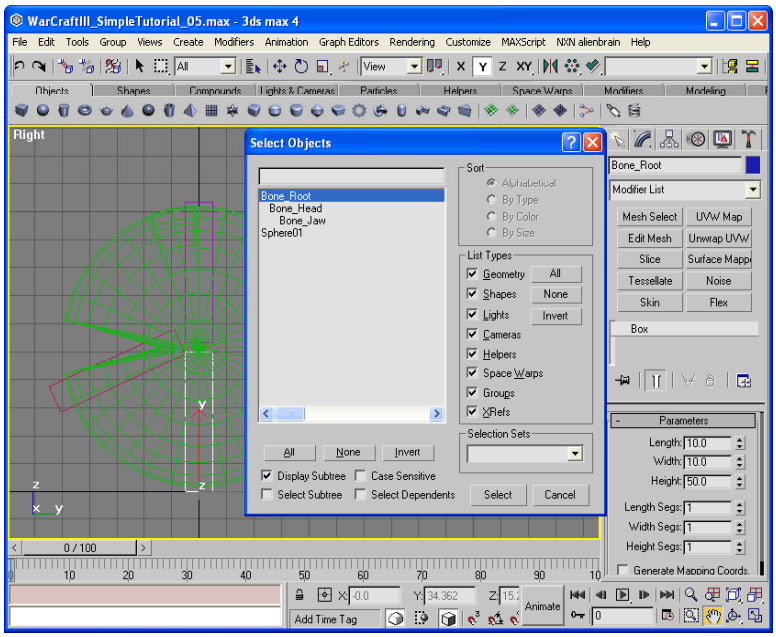

Figure 12

It is time to make a three-bone skeleton made of geometry. Make 3 boxes named Bone_Root, Bone_Head, and Bone_Jaw, respectively, and link them so Bone_Jaw is a child of Bone_Head, which is a child of Bone_Root. Place them so the root bone's origin is at the origin in 3ds max, and the jaw and head are both roughly at the center of the model (figure 12). Once you have the skeleton set up, make a key frame of position, rotation and scale for all the bones at frame 0. For now, link the Simple Unit's sphere geometry to the Bone_Root. We will use the other two bones in a few more steps.

Next we will animate the root bone for the walk cycle. Make the unit bounce with a few key frames starting at frame 10 and ending on frame 40, same as the walk sequence we made in Step 4. The quickest way we found to do this was by copying frame 0 to frame 10 and 40. On frame 25 with the Animate button on, we animated the root bone up along the y axis about 100 units. If you preview your work now, there will be a few warning errors, but you can see your animation. To fix the errors, we will need to understand the User Property Editor. Also let's look at skinning, called connected vertices in Warcraft III, before we continue the tutorial.

The User Property Editor

This rollout panel in the Utilities Toolbar is the interface for editing the user-defined object properties. This menu changes to show available options depending what is selected. If geometry is selected, the rollout will look like figure 13.

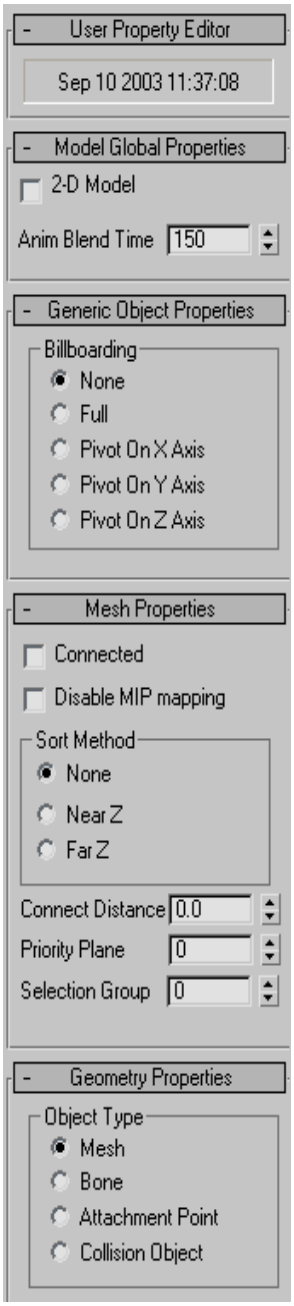

Figure 13

User Property Editor options

2-D Model – This is for making and previewing art to be displayed in the 2D game space like the UI cursor. When this is checked, the scale used by the game goes from 0 to 1 along the x and y axes (very small).

Anim Blend Time – This allows a preview of blend time from one animation to another. This value is not used in the game.

Generic Object Properties

Billboarding – Causes a bone to rotate procedurally with relation to the camera as if the front view in 3ds max was the camera direction. Full billboarding will cause the object to always face the camera. The other options allow locking of axes.

Mesh Properties

Connected – This option is for models to skin across multiple pieces of geometry. See Skinning the Model on the next page.

Sort Method – Alpha blend geosets can be made to sort by z distance against other alpha blend geosets.

Connect Distance – This is the maximum distance that a pair of connected vertices can be from each other and still be counted as connected. Zero is the default value, which actually equates to .25 units. Entering a value other than zero will specify an exact distance.

Priority Plane – Alpha-queued geosets can be made to draw in a specific order with relation to each other in the same model. The lower the value is, the sooner it is rendered. Values between -20 and 20 are regularly used.

Selection Group – Unused.

Object Types, Mesh – Default geometry.

Bone – Used to flag geometry so that it will not draw; instead, it will act as a skeleton with full transforms.

Attachment Point – Attachment objects are geometry type bones that don't draw but can have art procedurally attached. See Setting up Attachment Points.

Collision Object – For flagging hit test geometry that is not drawn.

Skinning the Model

Connecting vertices is a straightforward way to set up equal bone weighting between geometry pieces. Instead of having one geometry set influenced by many bones, the Warcraft III Art Tools require that the individual pieces be chopped up into different meshes in 3ds max. A leg can be made up of an upper leg and a lower leg. The vertices around the knee on both the upper and lower leg meshes must line up at frame 0, and both leg objects need to have the User Property – Connected flag checked. When they animate in the 3ds max viewport, they will cause holes to appear, but when they are set up correctly, the geometry will appear in the preview as welded together, causing the knee vertices to get 50/50% between the two parent leg bones. Up to four bones can influence one vertex.

Simple Unit Tutorial, step 6

It is time to cut the Simple Sphere Unit into three meshes. Using an Edit Mesh modifier with polygon selection, select the top half of the polygons, including the top of the mouth. Next, detach this selected region as a separate mesh named **Head***. Do the same for the* **Jaw,** *as in figure 14. While we are at it, let's rename the sphere to* **Body.**

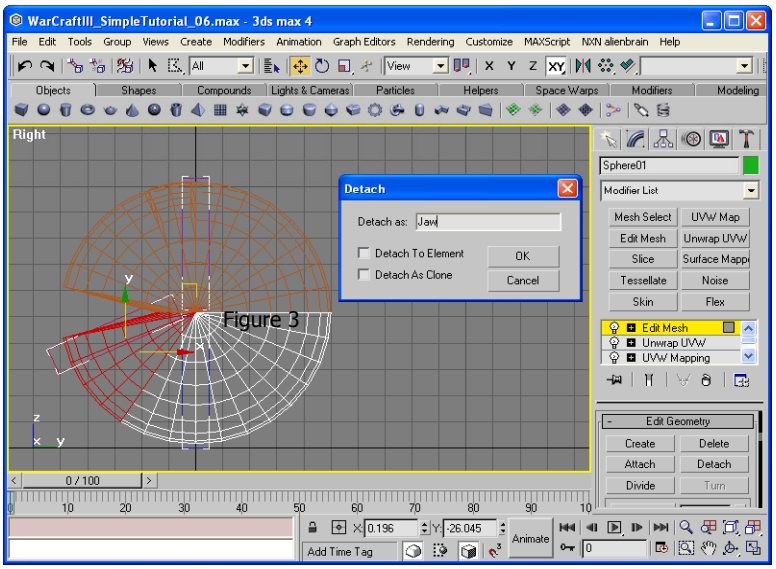

Figure 14

Now these three geometry meshes can be linked to their bone parents.

The bones need to be flagged as bones in the Utilities tab, User Property Editor. Additionally we need to set the Connected flag for all three meshes from the User Property Editor. To make sure everything works, we need to animate these bones. Go to the sequence of time you have set for the attack animation (in my file it is frames 110 through 140) and animate the jaw closing about halfway through the sequence, then opening again. I added a key at 5 frames in (frame 115) to close the mouth and a copy of that key at 15 frames in. Make sure you add keys for the beginning and the end of the sequence, and that key frame 0 still has the vertices lined up. Now use preview to look at what you have done so far. If you switch to the attack animation by pressing the "+" key a couple of times, the attack-biting animation will play once. To play it again, press the "" key. Here is the list of all preview commands.*

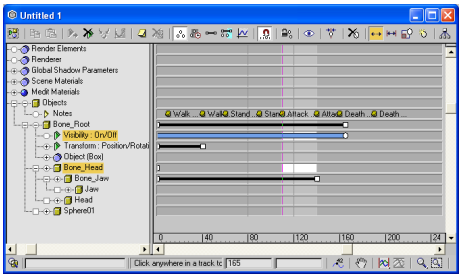

Figure 15

With the death animation, we need this unit to just disappear. In the track view, add a visibility track to the root bone. Make sure it is an On/Off controller type. Add a key to turn it off at the first 5 frames of the death sequence. (Figure 15) We will add a particle effect to this sequence in the next step.

Get creative and animate the stand animation however you like.

Warcraft III Particle System

Under the Create Toolbar – Particle Systems, you will find the BlizPart particle object. This plug-in is a complete particle system and material editor in one.

The particle type is always four vertex quads. There are three key frames you can set to determine the life of a particle: the initial Start, the Middle with user-placeable timing, and the End just as the particle life has ended. Animating the textures can then be done in the two segments: one from Start to Middle called *lifespan*, and the other from Middle to End called *decay*.

The Warcraft III particle system uses a relative orientation. Thus, it behaves oddly if you don't do the following: always make these objects in the top viewport and animate them into position, leaving them with 0,0,0 rotation at frame 0.

A gravity space warp is used to add gravity per emitter.

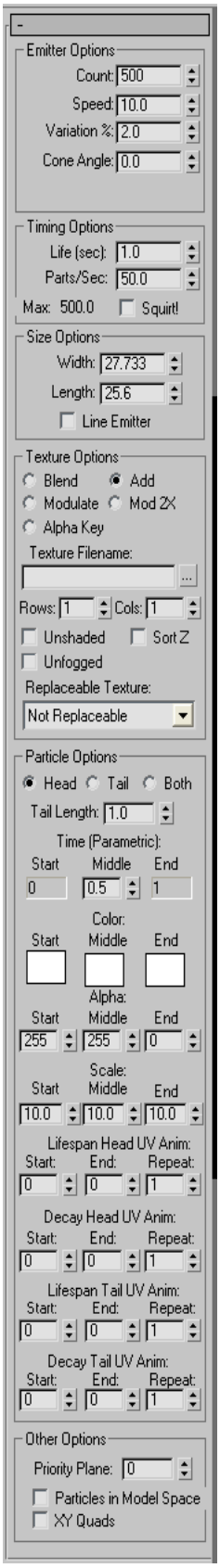

Figure 16

BlizParticle options

Count – Has no effect.
Speed – Initial velocity of created particles. Can be animated.
Variation % – Random variation % of speed.
Cone Angle – Angle of random spray pattern from center. A value of 180 will cause particles to emit in a 360-degree ball.

Timing options

Life – Number of seconds each particle continues to be drawn after its creation.
Parts/Sec – Number of particles created per second. Can be animated.
Squirt – When the Squirt option is checked and an animated stepped Bezier controller is used on parts/sec, a large number of particles will emit simultaneously on the parts/sec keys in the amount of the key value. This is good for explosion effects.
Width and Length – Width and length of the emitter plane where particles are created randomly from its surface area.
Line Emitter – Cause the cone spray pattern to lock to one axis.

Texture options

Blend, Add, Modulate, Mod, Mod 2X, and Alpha Key - Identical to the options in the Warcraft III material. Alpha Key is also known as 2 Bit Alpha.
Texture Filename – The .TGA texture being used for the particles.
Rows and Cols – If the particle is using sections of the texture at a time, you can use these options to set the number of rows and columns in a texture. Animating texture particles can be made to have frames all on one texture map that start at the top left and go to the right.
Unshaded – The particles do not use lighting and are always full bright.
Unfogged – Particles ignore distance fog.
Replaceable texture – Can be set to team glow to create procedurally colored glowing particles for a unit.

Particle options

Head – The particle is a billboarded square quad.
Tail – A tail particle is billboarded along the axis of motion and stretches in length based on speed.
Both – Draws both heads and tails.
Tail Length – A multiplier to the calculated tail particle length.
Time (parametric) Middle – The relative time of the middle key to the life.
Color, Start/Middle/End – Three values of RGB Multiply for the particle.
Alpha, Start/Middle/End – Three alpha values for the particle. Zero is transparent.
Scale, Start/Middle/End – Scale of the particle in units across.
Lifespan Head UV Anim – Starting and ending cell number for animating texture sequence of the first half of the Head particle's life. Repeat will cause this animating texture to cycle.
Decay Head UV Anim - Starting and ending cell number for animating texture sequence of the second half of the Head particle's life. Repeat will cause this animating texture to cycle.
Lifespan Tail UV Anim - Starting and ending cell number for animating texture sequence of the first half of the Tail particle's life. Repeat will cause this animating texture to cycle.
Decay Tail UV Anim - Starting and ending cell number for animating texture sequence of the second half of the Tail particle's life. Repeat will cause this animating texture to cycle.

Other options

Priority Plane – When you are using blend mode, the Priority Plane option can cause particles to draw in a specific order with other blend objects in the same model. The lower the number, the earlier the particles will draw.
Particles in Model Space – Causes animation of the particle emitter to be carried over to the particles.
XY Quads – Causes the particles to be a type of tail that orients to the XY grid.

Simple Unit Tutorial, step 7

Figure 17

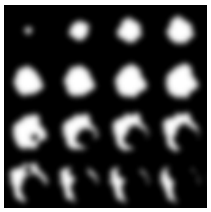

Figure 18

Our Sphere Unit just disappears in the death animation. I think we should make the unit explode by adding a squirt particle that will have animating smoke. We will be using a texture that has animating frames in rows and columns. You can either make one yourself or use the supplied texture CartoonSmoke.tga. The texture should be 128x128 with an alpha channel like the image in figures 17 & 18.

Select a BlizParticle from the Create/Particle System panel. From the top viewport, make a particle in the middle of our model. We will need to set up a few things in the Modify panel with the particle selected. Set the Cone Angle to 45. With Texture Options, select Blend and pick your texture. I will be using CartoonSmoke.tga. This texture has 4 rows and columns, so set Rows to 4 and Cols to 4. Set the scale to 50, 75, and 100 for Start, Middle, and End. Since the texture has multiple frames done with rows and columns, we need to set the Lifespan Head UV Anim to 0, 7, 1 for Start, End, and Repeat. The Decay Head UV Anim should be set to 8, 15, 1. The UV cell numbers refer to the location on the texture, starting with zero at the top left and going to the right.

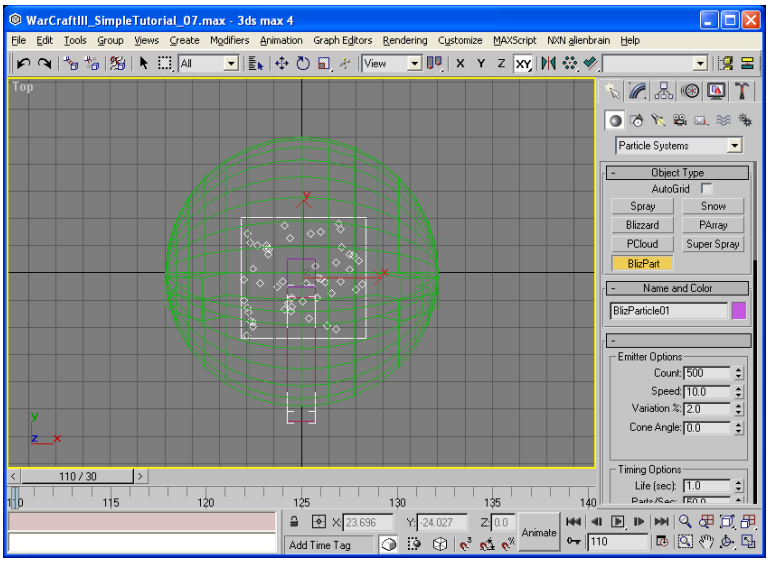

Figure 19

If you preview your work now, the particles will continually be emitting. We need to make them appear in a poof when the unit dies. The easy way to do this is to put a visibility track on the particle emitter with an On/Off controller and make it visible for a few frames of the death animation. Now you're ready to start changing things to fine-tune the effect. I changed the variation % to 90, and that made the animation look pretty cool. It's time for you to experiment.

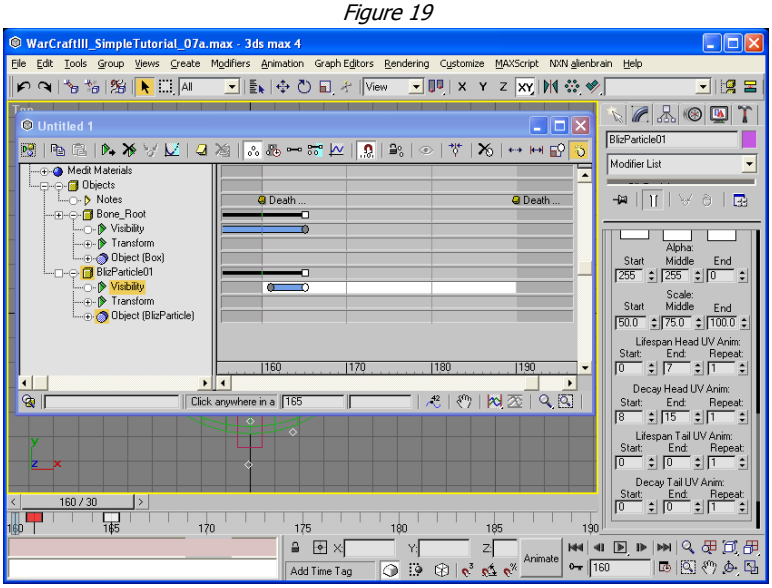

Figure 20

Ribbon Helpers

In the Warcraft III, many of the missile weapons and spells use a BlizRibbon helper. This object can be found under the Create\Helper tabs. A *ribbon* is simply a line that dumps out a sheet of quads wherever it moves. If it never moves, it can't be seen, despite the fact that it is creating quads.

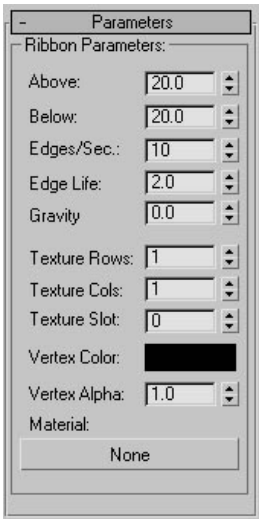

Figure 21

Ribbon helper options

Above and Below – These fields define the width of a ribbon in units based on their offset from the origin.
Edges/Sec – The number of quads generated.
Edge Life – Time in seconds that the quads stay around after being generated.
Gravity – Can be positive or negative. Will cause the ribbon to sink or rise in the z axis over time.
Texture Rows and Cols – Allows an animating texture similar to BlizParticle. Set the number of rows and columns equal to the texture.
Texture Slot – Pick the index number of rows and columns, and animate this number to get a cycle.
Vertex Color – An RGB multiple for the material.
Vertex Alpha – A fixed alpha value for the material.
Material – Use this to assign a Warcraft III material in the Material Editor to the Ribbon Object. It pulls from the material the diffuse texture, blend mode, and many other texture properties.

Setting up Attachment Points

To get spell effects in the game, units and buildings have attachment objects with specific names linked to the skeleton. These objects are boxes with the User Property – Attachment Point flagged. They don't render and don't need materials. They should not have any animation on their own.

Unit Attachment Point List	Building Attachment Point List
Chest Ref	Sprite RallyPoint Ref
Foot Left Ref	Sprite First Ref
Foot Right Ref	Sprite Second Ref
Hand Left Ref	Sprite Third Ref
Hand Right Ref	Sprite Fourth Ref
Head Ref	Sprite Fifth Ref
Origin Ref	Sprite Sixth Ref
Overhead Ref	Origin Ref
Weapon Ref	Overhead Ref

There is a full list of attachment names in Appendix B.

Event Objects

Certain visual and sound effects are triggered in the animation by event objects. These events can trigger splats and sounds, or they can spawn objects that are in the Warcraft III game. A spawned object is a point helper object with a specific name and a note track with keys. The keys need no info. The name of the point helper is an eight-digit code which the game looks up from one of three SLKs, then plays the appropriate sound or art. In the case of a Human Footman's right footprint, for instance, the point helper attachment would be named FPTxFBR1.

Footprints	FPTxyyyy	Appendix C
Blood Splats	SPLxyyyy	Appendix C
Uber Splat	UBRxyyyy	Appendix D
Sound Events	SNDxyyyy	Appendix E
Spawn Objects	SPNxyyyy	Appendix F

"x" represents any single character to make the object's name unique.
"yyyy" represents the four-character event object code listed in the Appendices.

Optimizing the Model

Warcraft III has a model detail setting in the options. This setting chooses between differing polygon count models that shipped with the game. User-created units can only add one polycount version. Here are guidelines that are Blizzard approved for High detail.

Unit Type	Poly Count	Texture size	Bones	Geosets*
Peon Class Unit	300	256k	18	3
Medium-sized Unit	375	256k	22	3
Big Units or Mounted	500	512k	30	5
Uber Unit (something really big)	525	1MB	30	5
Small Building	300	256k	5	3
Medium Building	600	512k	5	3
Town Hall Building	800	1MB	15	5
Destructible Tree	75	256k	1	1

*Geoset counts are for walk and stand animations. Effects like death can be higher.

Using Warcraft III Preview

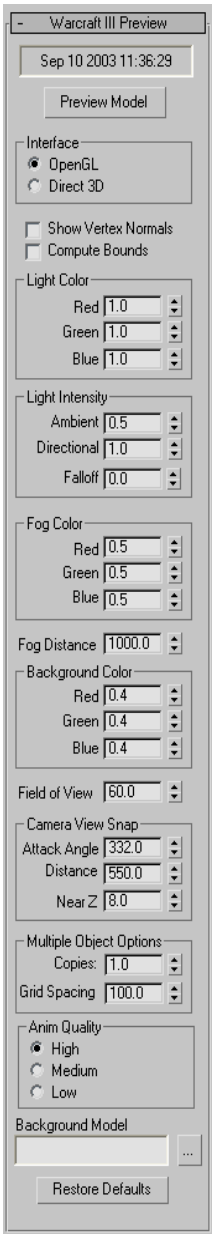

Figure 22

The Warcraft III Previewer enables you to see a very close representation of what the model will look like in the game. You can select the Warcraft III Previewer from the Utilities rollout panel in 3ds max. The Warcraft III Previewer has its own rollout, with many options.

The first button on the rollout is the "Preview Model" button. This is the most frequently used selection on the rollout panel. Pressing "Preview Model" causes an external window to open that emulates what the model will look like in Warcraft III. The other options on the rollout affect what the preview window looks like.

Warcraft III Preview options

Interface: You can use either OpenGL or Direct3D to view the model preview.
Show Vertex Normals: If this box is checked, small lines will be projected perpendicular to the surface. The normals will be shown at the vertices. If a vertex belongs to multiple faces, you will see multiple normals at the vertex, one for each face that the vertex belongs to.
Compute Bounds: If this box is checked, the Previewer will compute bounding information for the model. This takes extra time, especially on complicated models with multiple animation sequences. Note that the bounds are automatically calculated when the model is exported. You can view the bounds using the "B" command key in the Previewer. See Preview Commands for more information.
Light Color: Use the spinner controls to select RGB light values. Each component must be in the range 0.0 to 1.0.
Light Intensity: You can use the spinner controls to set different values for ambient and directional lights, and the falloff parameter for the directional light. The ambient and directional values must be in the range of 0.0 to 1.0. The falloff parameter is divided by 100000 and used as the quadratic lighting falloff parameter.
Fog Color: Use the spinner to set the RGB value for the fog color.
Fog Distance: This is the distance at which the fog value will be 100%, in world units.
Background Color: Use the spinners to set the RGB for the background color of the display window.
Field of View: This value represents the camera in-game field of view, in degrees. The default value is very similar to what Warcraft III uses in the game.
Camera View Snap: These values set the default camera position when the preview window opens. They are not generally changed from the default values.
Multiple Object Options: You can set the value of Copies and Grid Spacing to preview multiple copies of the model at one time. This is frequently useful for performance testing; in these cases it's not uncommon to preview 100 copies at once. If you enter a fractional value for Copies, it will be truncated to an integer value. The grid spacing is the number of world units between each copy; you may need to set this to a larger value for larger units.
Anim Quality: These settings represent the same settings as the Warcraft III in-game animation quality settings. You can use these buttons to preview what your animations will look like in different quality settings in the game. High quality is the default; no processing is done to the animation. Medium quality has all TCB controllers turned into linear controller. Low quality has all TCB controllers turned into linear controller and no motion blending applied.
Background Model: You can specify the filename of another MDL or MDX that will be shown at the origin during your model preview. This can be useful for testing relative scale of a model.
Restore Defaults: Resets all of these fields to their default installation values.

Warcraft III Preview Commands

These keyboard commands are available when the Previewer is open:
Left Mouse + Drag: Orbit the camera around the model.
Right Mouse + Drag: Zoom in or zoom out the camera.
Control + Left Mouse + Drag: Orbit the light direction around the model.
Left and Right Arrow: Strafe the camera left or right.
Up and Down Arrow: Strafe camera up or down.
Space bar: Pause/Resume animation.
Page Down: Advance animation one frame.
Page Up: Reverse animation one frame (NB: Particle systems DO NOT run in reverse).
Home: Resets current sequence timing.
L: Toggle lighting.
X: Toggle graph display.
Keypad +: Next sequence.
Keypad -: Previous sequence.
Keypad *: Reset this sequence to the beginning.
P: Toggle text displays.
F: Toggle fog.
F1: Slow down time scale.
F2: Speed up time scale.
F3: Reset time scale to normal time.
M: Toggle display of mipmap levels.
B: Cycle through bounding volume displays: sphere, box, none.
V: Reset view to default position.
C: Toggle between default camera and scene camera, if one exists.
Q: Toggle look-at. If you have a bone named "bone_head", it will look at the light position. Otherwise, the first bone will look at that position.
A: Add the current sequence to the queue of sequences to playback.
G: Clear all sequences from the queue of sequences to playback.
S: Toggle playback of all sequences in the queue of sequences.
1: Decrease vertex alpha.
2: Increase vertex alpha.
3: Decrease animation blend time.
4: Increase animation blend time.
0: Toggle vertex color between white and red.
6: Toggle display of collision geometry.
9: Toggle display of unselectable geometry.
F8: Cycle through particle levels of detail (high, medium, low). This affects the number of particles that the system can emit (100%, 65%, 30%).
F9: Cycle through the light levels of detail. The affects the maximum number of lights in the scene (Hardware-limited, 4, or 1).
[: Select the previous replaceable texture for this category.
] : Select the next replaceable texture for this category.
Ctrl + [: Select the previous replaceable texture category.
Ctrl +] : Select the next replaceable texture category.

Exporting the Model From 3ds max

After you've previewed your model, optimized it, and debugged it, you can export it for use in the game. Make sure to save your work first.

Choose File->Export from the list. You will see the "Select File to Export" dialog. Choose "Warcraft III Model" from the "Save as type" combo selector. If you do not choose a file extension for your model, it will be exported in binary format with the file extension .MDX. Otherwise, if you enter a filename that ends in .MDL, the model will be exported in text format. You should not use the text format except for debugging purposes. After you enter the filename and click "Save", the Warcraft III Art Tools will convert the model data into the correct format. If there are problems with the model, you will be presented with warnings for the model. If there are errors that would prevent the model from working, you will receive an error message, and the export will not continue. After the data are processed, the model file is saved, and all of the textures used by the model are converted to "blip" format (.BLP file extension) and saved in the same directory as the model. The model is now ready to use in the game.

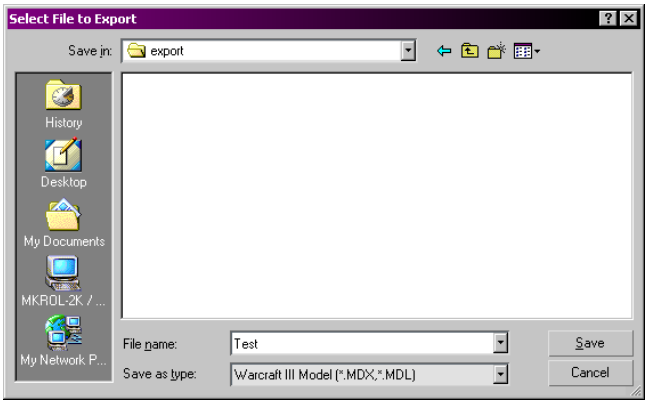

Figure 23

Importing the Model into a Map Using WorldEdit

Once the model and textures have been exported to files on disk, they can be imported for use in maps or campaigns via the Warcraft III World Editor, which comes with the game. This is done using the Import Manager module (*Module->Import Manager*) for maps, or the Imported Files tab within the Campaign Editor module (*Module->Campaign Editor*) for campaigns. Note that you must have The Frozen Throne expansion pack to access the Import Manager and Campaign Editor modules.

Import the model and textures files by using the *File->Import File(s)* command in the Import Manager (*Edit->Import File(s)* in the Campaign Editor), and selecting the file or folder to import. Once the import is complete, you will need to adjust the path of the texture files to match the way they are referenced from within the model file. By default, all files are imported to the *war3mapImported* directory for map files, or *war3campImported* for campaign files. This can be changed by modifying the properties of the imported file via *Edit->Modify File Properties*, or simply double-clicking the file in the list. In the resulting *Imported File Properties* dialog, check the box titled *Use custom path* and enter the path to the texture file as it is referenced from within the model file. This often means simply deleting the extraneous *war3mapImported* or *war3campImported* directory name from the path.

After you import the model and texture files, we recommend that you save your map or campaign, then close and reload it. This will ensure that the editor has loaded your new model and textures and will display them properly when they are subsequently used.

You can then select your model file from the *Import* menu in any model selection dialog. This dialog is used when modifying model fields in the Object Editor (such as *Art – Model File* for units), as well as parameters for certain trigger functions in the Trigger Editor (such as the *Special Effect* actions).

Optional: Creating Model Portraits

Units can have a separate model for a portrait. This new art model will have the suffix "_Portrait.mdx". Example: Footman.mdx and Footman_Portait.mdx. A portrait needs a target camera in the model.

Doodads and Destructibles

Warcraft III can have multiple MDX files as variations of the same type of tree, and this multi-file approach works for any destructible or doodad. Up to 10 variations can be made by adding a one-digit number suffix starting with 0 (i.e., **newtree0.mdx**, **newtree1.mdx**, etc.).

Trees are set up as Simple Destructible objects so they can draw many on screen very quickly. For each tree the game will read three different MDX files: one with the standing tree and no animation, one with the dead version with no animation, and one version with all of its animation. These files will be named by adding an **S** for stand and a **D** for dead. When the letters are added, the above variation names will therefore become **newtree0S.mdx**, **newtree0D.mdx**, **newtree0.mdx.**

Reference Information

Animation List

This is a reference list of many animation name combinations used in Warcraft III.

All Units	**Description and Notes Track Info**
Stand	2 or more anims, varying lengths, all start and stop on the same Stand Position. 1 with very subtle anim and the other "fidgets" with a rarity of 1 or greater
Walk	1 with Move Speed
Attack	2 attacks, the impact time should be less than 1 second, start and stop in the Attack Position, NonLooping
"Stand Ready"	A Looping animation between attacks. Starts and stops with Attack Position
Death	Starts with Ready and ends lying on the ground, NonLooping
"Decay Flesh"	Starts from Death and fades to Bones smoothly over the whole anim, NonLooping, 1800 Frames
"Decay Bone"	Starts from Decay Flesh and fades to nothing at the very end of the anim, NonLooping, 1800 Frames
Spell	Generic spell cast, NonLooping
Hero Units	
Dissipate	Hero units only; use this anim after Death, NonLooping
"Stand Channel"	Looping spell casting animation
Summonables	
Birth	From nothing to Stand Position, NonLooping, 1800 Frames
Creeps	
"Spell Slam"	Generic spell cast, NonLooping
Amphibious	
Stand Swim"	2 or more anims, varying lengths, all start and stop on the same Stand Position. 1 with very subtle anim and the other "fidgets" with a rarity of 1 or greater
"Walk Swim"	1 with Move Speed (the same speed as the land Walk)
"Attack Swim"	2 attacks, the impact time should be less than 1 second, start and stop in the Attack Position, NonLooping
"Stand Ready Swim"	A Looping animation between attacks. Starts and stops with Attack Position
"Death Swim"	Starts with Ready and ends lying on the ground, NonLooping
"Decay Swim"	Starts from Death and fades to nothing smoothly over the whole anim, NonLooping, 1800 Frames
"Spell Swim"	Generic spell cast, NonLooping
Hero Amphibious Units	
"Dissipate Swim"	Hero units only; use this anim after Death, NonLooping
"Stand Channel Swim"	Looping spell casting animation
Amphibious Creeps	
"Spell Slam Swim"	Generic spell cast, NonLooping
Portrait Models	
"Portrait"	2 or more animations, 1 with very subtle anim and the other "fidgets" with a rarity of 1 or greater
"Portrait Talk"	2 or more animations
Morphing Units	
Morph	Stand to "Stand Alternate" (for units with two forms), NonLooping
"Morph Alternate"	"Stand Alternate" to Stand, NonLooping
"Stand Alternate"	
"Stand Ready Alternate"	
"Attack Alternate"	NonLooping
"Stand Work Alternate"	
"Walk Alternate"	
"Death Alternate"	Goes to the Regular "Decay Flesh", NonLooping
NonRaiseable	
Decay	For units that don't make "fleshy" corpses, from death to nothing, NonLooping,
Peons	
"Stand Work"	Looping work anim
"Walk Gold"	
"Stand Gold"	
"Stand Work Gold"	
"Attack Gold"	
Buildings	
Birth	NonLooping, Time Scaled
Stand	
Stand Work	
Death	NonLooping
Decay	Generic decay regardless of upgrade (use external ref decay per building size), NonLooping, Time Scaled
Portrait	
Upgraders	

Birth Upgrade First	NonLooping, Time Scaled
Stand Upgrade First	
Stand Work Upgrade First	
Death Upgrade First	NonLooping
Portrait Upgrade First	
Birth Upgrade Second	NonLooping, Time Scaled
Stand Upgrade Second	
Stand Work Upgrade Second	
Death Upgrade Second	NonLooping
Portrait Upgrade Second	
Destructibles	
Birth	NonLooping, Time Scaled
Stand	
Death	Stumps are left in the end of the death anim, NonLooping
"Stand Hit"	A short anim denoting damage currently being taken, NonLooping
Spells and Missiles	
Birth	
Stand	
Death	NonLooping
Other	
Death Explode	
Walk Fast	

Replaceable Texture ID

In Warcraft III, certain textures are marked as *replaceable textures*. These textures are swapped out at run time with one of a set of textures. The exact texture that is substituted for the original texture applied to the model is determined by the game at run time. An example of this is team color. For any model with a texture marked with the Replaceable ID "Team Color", that texture will be replaced by the current team color for the team in which the model belongs. This technique means that one model can represent all 16 teams in the game without requiring different variations. You can use the Model Previewer to view the replacement textures in place on the model; the Previewer replaces the textures in exactly the same manner Warcraft III does.

The Replaceable Texture IDs that you can choose from are these:

Team Color – This is the current player's team color. In multiplayer maps, the computer selects this color. Usually, an "underpainting" texture is applied to part of the model or to the entire model, and this texture is set to team color. Then the model's skin is applied with alpha "holes", where the team color should show through the skin. You can see this technique in the sample models.

Team Glow – This is a square texture that has a color "hotspot" at the center, fading in a circular gradient to complete transparency by the edges. This is frequently applied to billboarded quads on Hero units to make the Hero or the Hero's weapon glow. The game will always set team color and team glow to the same color. See the sample Hero model for an example of this technique.

Trees – The game will replace textures marked with Tree with tree textures that match the current tileset. The game can also replace the tree texture with blighted versions of the tree texture that match the current tileset.

Appendix A - Animation Tokens

Primary Animations	Animation Properties
attack	alternate
birth	alternateex
cinematic	chain
death	channel
decay	complete
dissipate	critical
morph	defend
portrait	drain
sleep	eattree
spell	fast
stand	fill
walk	flail
	flesh
	fifth
	fire
	first
	five
	four
	fourth
	gold
	hit
	large
	left
	light
	looping
	lumber
	medium
	moderate
	off
	one
	puke
	ready
	right
	second
	severe
	slam
	small
	spiked
	spin
	swim
	talk
	third
	three
	throw
	two
	turn
	victory
	work
	wounded
	upgrade

Appendix B - Bone and Link Point Reference

PRIMARY BONES		
CHEST	bone_chest	
FOOT	bone_foot	unused
HAND	bone_hand	unused
HEAD	bone_head	
TURRET	bone_turret	
BONE QUALIFIERS		
ALTERNATE	alternate	
LEFT	left	unused
MOUNT	mount	unused
RIGHT	right	unused
SMART	smart	unused
PRIMARY LINK POINTS		
CHEST	chest	
FEET	feet	
FOOT	foot	
HAND	hand	
HEAD	head	
ORIGIN	origin	
OVERHEAD	overhead	
SPRITE	sprite	(for building attachment points)
WEAPON	weapon	
LINK POINT QUALIFIERS		
ALTERNATE	alternate	
LEFT	left	
MOUNT	mount	
RIGHT	right	
REAR	rear	
SMART	smart	
FIRST	first	(for building attachment points)
SECOND	second	(for building attachment points)
THIRD	third	(for building attachment points)
FOURTH	fourth	(for building attachment points)
FIFTH	fifth	(for building attachment points)
SIXTH	sixth	(for building attachment points)
SMALL	small	(for building attachment points)
MEDIUM	medium	(for building attachment points)
LARGE	large	(for building attachment points)
GOLD	gold	
RALLY	rallypoint	(for building attachment points)
EATTREE	eattree	

Appendix C - Splat/Footprint Data

DBL0	DemonBloodLarge0
DBL1	DemonBloodLarge1
DBL2	DemonBloodLarge2
DBL3	DemonBloodLarge3
DBS0	DemonBloodSmall0
DBS1	DemonBloodSmall1
DBS2	DemonBloodSmall2
DBS3	DemonBloodSmall3
EBL0	NightElfBloodLarge0
EBL1	NightElfBloodLarge1
EBL2	NightElfBloodLarge2
EBL3	NightElfBloodLarge3
EBS0	NightElfBloodSmall0
EBS1	NightElfBloodSmall1
EBS2	NightElfBloodSmall2
EBS3	NightElfBloodSmall3
FAL0	FootprintBareSmallLeft
FAL1	FootprintBareLargeLeft1
FAL2	FootprintBareGiantLeft1
FAL3	FootprintBareHugeLeft1
FAR0	FootprintBareSmallRight1
FAR1	FootprintBareLargeRight1
FAR2	FootprintBareGiantRight1
FAR3	FootprintBareHugeRight1
FBL0	FootprintBootSmallLeft0
FBL1	FootprintBootLargeLeft0
FBL2	FootprintBootSmallLeft1
FBL3	FootprintBootLargeLeft1
FBL4	FootprintBootGiantLeft1
FBR0	FootprintBootSmallRight0
FBR1	FootprintBootLargeRight0
FBR2	FootprintBootSmallRight1
FBR3	FootprintBootLargeRight1
FBR4	FootprintBootGiantRight1
FCR0	FootprintClovenSmallRight
FCL0	FootprintClovenSmallLeft
FCL1	FootprintClovenLargeLeft
FCR1	FootprintClovenLargeRight
FCL3	FootprintClovenXtraLargeLeft
FCR3	FootprintClovenXtraLargeRight
FCL2	FootprintClovenReallySmallLeft
FCR2	FootprintClovenReallySmallRight
FHL0	FootprintHorseSmallLeft
FHL1	FootprintHorseLargeLeft
FHR0	FootprintHorseSmallRight
FHR1	FootprintHorseLargeRight
FPL0	FootprintPawLeft0
FPL1	FootprintPawLargeLeft

FPR0	FootprintPawRight0
FPR1	FootprintPawLargeRight
FRL0	FootprintRootLeft
FRR0	FootprintRootRight
FRL1	FootprintRootLeftSmall
FRR1	FootprintRootRightSmall
FTL0	FootPrintTrollLeft
FTR0	FootPrintTrollRight
FWL0	FootPrintWheelLeft0
FWR0	FootPrintWheelRight0
FWL1	FootPrintWheelLeft1
FWR1	FootPrintWheelRight1
FML0	FootPrintMurlocLeft
FMR0	FootPrintMurlocRight
FPL2	FootPrintPawBearLeft
FPR2	FootPrintPawBearRight
FSL0	FooprintSkeletonRight
FSR0	FooprintSkeletonLeft
FSL1	FooprintSkeletonGiantRight
FSR1	FooprintSkeletonGiantLeft
FFL0	FootprintFlameLeft
FFR0	FootprintFlameRight
FFL1	FootprintFlameGiantLeft
FFR1	FootprintFlameGiantRight
FKL0	FootprintSpikeLeft
FKR0	FootprintSpikeRight
FKL1	FootprintSpikeLargeLeft
FKR1	FootprintSpikeLargeRight
FKL2	FootprintSpikeGiantLeft
FKR2	FootprintSpikeGiantRight
FLSL	FootprintLizzardSmallLeft
FLSR	FootprintLizzardSmallRight
FLLL	FootprintLizzardLargeLeft
FLLR	FootprintLizzardLargeRight
FSSL	FootprintSnakeSmallLeft
FSSR	FootprintSnakeSmallRight
FSLL	FootprintSnakeLargeLeft
FSLR	FootprintSnakeLargeRight
FDSL	FootprintDragSmallLeft
FDSR	FootprintDragSmallRight
FDLL	FootprintDragLargeLeft
FDLR	FootprintDragLargeRight
HBL0	HumanBloodLarge0
HBL1	HumanBloodLarge1
HBL2	HumanBloodLarge2
HBS0	HumanBloodSmall0
HBS1	HumanBloodSmall1
HBS2	HumanBloodSmall2

HBS3	HumanBloodSmall3
HBL3	HumanBloodLarge3
OBL0	OrcBloodLarge0
OBL1	OrcBloodLarge1
OBL2	OrcBloodLarge2
OBL3	OrcBloodLarge3
OBS0	OrcBloodSmall0
OBS1	OrcBloodSmall1
OBS2	OrcBloodSmall2
OBS3	OrcBloodSmall3
UBL0	UndeadBloodLarge0
UBL1	UndeadBloodLarge1
UBL2	UndeadBloodLarge2
UBL3	UndeadBloodLarge3
UBS0	UndeadBloodSmall0
UBS1	UndeadBloodSmall1
UBS2	UndeadBloodSmall2
UBS3	UndeadBloodSmall3
WSL0	WaterSplashLarge0
WSL1	WaterSplashLarge1
WSS0	WaterSplashSmall0
WSS1	WaterSplashSmall1
WHL0	WaterHumanBloodLarge0
WHL1	WaterHumanBloodLarge1
WHS0	WaterHumanBloodSmall0
WHS1	WaterHumanBloodSmall1
WOL0	WaterOrcBloodLarge0
WOL1	WaterOrcBloodLarge1
WOS0	WaterOrcBloodSmall0
WOS1	WaterOrcBloodSmall1
WEL0	WaterNightElfBloodLarge0
WEL1	WaterNightElfBloodLarge1
WES0	WaterNightElfBloodSmall0
WES1	WaterNightElfBloodSmall1
WUL0	WaterUndeadBloodLarge0
WUL1	WaterUndeadBloodLarge1
WUS0	WaterUndeadBloodSmall0
WUS1	WaterUndeadBloodSmall1
WDL0	WaterDemonBloodLarge0
WDL1	WaterDemonBloodLarge1
WDS0	WaterDemonBloodSmall0
WDS1	WaterDemonBloodSmall1
WSX0	WaterSplashOnly0
WSX1	WaterSplashOnly1

Appendix D - Uber-Splat Data

LSDS	LordSummerPlainDirtSmall
LSDM	LordSummerPlainDirtMed
LSDL	LordSummerPlainDirtLarge
HCRT	HumanCrater
UDSU	UndeadUberSplat
DNCS	DeathNeutralCityBuildingSmall
HMTP	HumanMassTeleport
SCTP	ScrollOfTownPortal
AMRC	AmuletOfRecall
DRKC	DarkConversion
DOSB	DeathOrcSmallBuilding
DOMB	DeathOrcMedBuilding
DOLB	DeathOrcLargeBuilding
DHSB	DeathHumanSmallBuilding
DHMB	DeathHumanMedBuilding
DHLB	DeathHumanLargeBuilding
DUSB	DeathUndeadSmallBuilding

DUMB	DeathUndeadMedBuilding
DULB	DeathUndeadLargeBuilding
DNSB	DeathNightElfSmallBuilding
DNMB	DeathNightElfMedBuilding
DNSA	DeathNightElfSmallAncient
DNMA	DeathNightElfMedAncient
HSMA	HumanUberSplat
HMED	HumanUberSplat.
HLAR	HumanUberSplat.
OSMA	OrcUberSplat
OMED	OrcUberSplat
OLAR	OrcUberSplat
USMA	UndeadUberSplat
UMED	UndeadUberSplat
ULAR	UndeadUberSplat
ESMA	AncientUberSplat
EMDA	AncientUberSplat

ESMB	NightElfUberSplat
EMDB	NightElfUberSplat
HTOW	TownHallUberSplat
HCAS	CastleUberSplat
NGOL	GoldmineUberSplat
THND	ThunderClap
NDGS	DemonGateSplat
CLTS	ThornyShieldSplat
HFS1	HumanFlameStrike1
HFS2	HumanFlameStrike1
USBR	Burrow
NLAR	NagaUberSplat
NMED	NagaUberSplat
DPSW	Dark Portal SW splat
DPSE	Dark Portal SE splat

Appendix E - Sound Data

Sound Event Code	Sound
FBCL	TestFootstep
FBCR	TestFootstep
AAMS	AntiMagicshell
AAST	AncestralSpirit
AAVE	ObsidianStatueMorph
ABLO	Bloodlust
ABPD	BothGlueScreenPopDown
ABPU	BothGlueScreenPopUp
ABRW	Burrow
ABSK	BerserkerRage
ABTR	BattleRoar
ACAN	Cannibalize
ACBC	BreathOfFire
ACBF	BreathOfFrost
ACCV	CrushingWave
ACWD	CrushingWaveDamage
ACLB	CinematicLightningBolt
ACRH	CorrosiveBreathMissileHit
ACRI	Cripple
ACRL	CorrosiveBreathMissileLaunch
ACRS	Curse
ACSI	Silence
ACSL	CreepSleep
ACYB	CycloneBirth
ACYD	CycloneDeath
ADEF	Defend
FDFL	DeepFootstep
FDFR	DeepFootstep
FDSL	FiendStep
FDSR	FiendStep
FHCL	HeroCinematicStep
FHCR	HeroCinematicStep
ADCM	DruidOfTheClawMorph
ADCA	DruidOfTheClawMorphAlternate
ADEV	Devour
ADHM	DemonHunterMorph
ADIS	DispelMagic
ADTM	DruidOfTheTalonMorph
ADTA	DruidOfTheTalonMorphAlternate
ADVP	DevourPuke
AEAT	EatTreeMunch
AEBA	Barkskin
AEBD	Earthbind

AEBL	BlinkCaster
AEBT	BlinkTarget
AHDR	SiphonManaCaster
AHEA	Heal
AHER	LevelUp
AHFS	FlameStrike
AHFT	FlameStrikeTarget
AHHB	HolyBolt
AHMC	MarkOfChaos
AHMF	ManaFlareMissile
AHMT	MassTeleport
AHRE	Resurrect
AHRV	ReviveHuman
AHTB	StormBolt
AHSL	StormBoltLaunch
AHTC	ThunderClap
AHWD	HealingWardBirth
AICB	OrbOfCorruptionLaunch
AICH	OrbOfCorruptionHit
AIDC	NeutralizationWandHit
AILL	ItemIllusion
AIMA	ManaPotion
AINB	InfernalBirth
AINF	InnerFire
AIRE	RestorationPotion
AISO	SoulGem
AITM	Tome
AIVS	Invisibility
AKDL	KodoDrumLeft
AKDR	KodoDrumRight
ALPD	LeftGlueScreenPopDown
ALPU	LeftGlueScreenPopUp
ALSD	LightningShield
ANBA	BlackArrowHit
ANDO	DoomTarget
ANDT	RevealMap
ANEU	NeutralBuildingActivate
ANHT	HowlOfTerror
ANMO	MonsoonBolt
ANPA	Parasite
ANSA	SacrificeUnit
ANSD	StrongDrink
ANSM	StrongDrinkMissile
ANSS	SpellShieldAmulet
AOAG	WardBirth

AOCR	CriticalStrike
AOHW	HealingWaveTarget
AOLB	LightningBolt
AOMI	MirrorImageDeath
AOMC	MirrorImage
AORE	Reincarnation
AORV	ReviveOrc
AOSD	FeralSpiritDone
AOSF	FeralSpiritTarget
AOSH	ShockWave
AOVD	VoodooBirth
AOWS	Warstomp
AOWW	Whirlwind
APHS	PhaseShift
APHX	PhoenixBirth
APLA	PolymorphAir
APLD	PolymorphDone
APOL	Polymorph
APRG	Purge
APSH	PossessionMissileHit
APSL	PossessionMissileLaunch
APXB	PhoenixEggBirth
AREJ	Rejuvenation
AREP	Repair
AROO	Root
ARPD	RightGlueScreenPopDown
ARPU	RightGlueScreenPopUp
ASHP	ShadowPact
ASKA	RaiseSkeletonArcher
ASKW	RaiseSkeletonWarrior
ASLC	SlowCaster
ASLO	Slow
ASPL	SpiritLink
ASPM	SpellStealMissileLaunch
ASPS	SpellStealTarget
ASTO	StoneFormMorph1
AST2	StoneFormMorph2
AST3	StoneFormMorph3
ASTA	StoneFormMorphAlternate
ASTB	StasisTotemBirth
ASTH	StampedeHit
ASTS	StasisTotemDeath
ASWB	SpiritWolfBirth
ASWE	WaterElementalBirth
ATAU	Taunt

ATRB	TreeWallBirth
AUB1	UndeadBuildingBirth1
AUB2	UndeadBuildingBirth2
AUB3	UndeadBuildingBirth3
AUB4	UndeadBuildingBirth4
AUCB	ScarabBirth
AUCD	CarrionSwarmDamage
AUCH	Charm
AUCO	UnstableConcoction
AUCS	CarrionSwarmLaunch
AUDA	DarkRitual
AUDC	DeathCoil
AUDM	DarkSummoningMissileLaunch
AUDP	DeathPactTarget
AUDS	DarkSummoningTarget
AUDT	DeathAndDecayTarget
AUFA	FrostArmor
AUFN	FrostNova
AUGS	GatherShadowsMorph
AUGA	GatherShadowsMorphAlternate
AUHF	UnholyFrenzy
AUIH	ImpaleLand
AUIM	Impale
AUIT	ImpaleHit
AUPR	Uproot
AURV	ReviveUndead
AWBS	BigWaterStep
AWEB	Web
AWRS	Pulverize
AWST	WaterStep
DABA	AbominationAlternateDeath
DABO	AbominationDeath
DACO	AcolyteDeath
DADR	DruidOfTheTalonDeath
DALB	AlbatrossDeath
DAMG	HeroArchMageDeath
DANG	WardDeath
DANP	TreantDeath
DARC	ArachnathidDeath
DARG	ArmorGolemDeath
DART	ArtilleryExplodeDeath
DASS	AssassinDeath
DWTC	WatcherDeath
DBAL	BallistaDeath
DBAN	BansheeDeath
DBAT	BatRiderDeath
DBES	HeroBloodElfDeath
DBLA	HeroBladeMasterDeath
DBNT	BanditDeath
DBRG	DeathBridge
DBRI	BristleBackDeath
DBSF	BlackStagFemaleDeath
DBSM	BlackStagMaleDeath
DBSP	BattleShipDeath
DBSX	ObsidianAvengerDeath
DBTM	BeastmasterDeath
DCAT	CatapultDeath
DCBL	DeathCityBuilding
DCEN	CentaurDeath
DCNA	CentaurArcherDeath
DCRL	HeroCryptLordDeath
DDEM	HeroDemonHunterDeath
DDHK	DragonHawkDeath
DDKN	HeroDeathKnightDeath
DDKR	DarkRangerDeath
DDMA	HeroDemonHunterDeathAlternate
DDMG	DoomGuardDeath
DDOC	DruidOfTheClawDeath
DDCA	DruidOfTheClawDeathAlternate
DDMN	DemonessDeath

DDNW	DuneWormDeath
DDTA	DruidOfTheTalonDeathAlternate
DDRA	DragonDeath
DDRL	HeroDreadLordDeath
DDRN	DraeneiDeath
DDRS	DragonspawnDeath
DDRY	DryadDeath
DDSH	HumanDissipate
DDSN	NightElfDissipate
DDSO	OrcDissipate
DDWF	DireWolfDeath
DEBA	DeathWalkingNightElfBuilding
DEBC	DeathNightElfBuildingCancel
DEGS	EggSackDeath
DELB	DeathNightElfLargeBuilding
DELS	DeathNightElfSmallBuilding
DENT	EntDeath
DFAR	HeroFarSeerDeath
DFCO	FacelessOneDeath
DFDR	FaerieDragonDeath
DFLG	FelguardDeath
DFEL	FelhoundDeath
DFOO	FootmanDeath
DFOR	ForgottenOneDeath
DFRG	FrogDeath
DFRM	FrostmourneDeath
DFRT	ForestTrollDeath
DFRW	FrostWyrmDeath
DFSP	ForestTrollShadowPriestDeath
DFTN	ForgottenOneTentacleDeath
DFUR	FurbolgDeath
DGAR	GargoyleDeath
DGAS	GargoyleStoneDeath
DGAT	GateDeath
DGHO	GhoulDeath
DGLD	GoldMineDeath
DGLM	GoblinLandMineDeath
DGNA	GnollArcherDeath
DGNL	GnollDeath
DGOB	GemstoneObeliskDeath
DGRU	GruntDeath
DGRY	GryphonRiderDeath
DGRZ	GrizzlyBearDeath
DGRS	BearSwimDeath
DGSD	GoblinSapperDeath
DGSP	GoblinSapperExplode
DGST	GiantSeaTurtleDeath
DTDS	GiantSeaTurtleDeathSwim
DGYR	GyrocopterDeath
DGZP	GoblinZeppelinDeath
DHBC	DeathHumanBuildingCancel
DHIP	HippogryphDeath
DHLB	DeathHumanLargeBuilding
DHLS	DeathHumanSmallBuilding
DHMC	HermitCrabDeath
DHOR	HorseDeath
DHRP	HarpyDeath
DHUN	HeadHunterDeath
DHWD	HealingWardDeath
DHDA	HydraDeath
DHDS	HydraDeathSwim
DHYD	HydraliskDeath
DICT	IceTrollDeath
DINF	InfernalDeath
DINM	InfernalMachineDeath
DIPW	PrisonWagonDeath
DIRG	IronGolemDeath
DJAN	JainaDeath
DKBS	KoboldShovelerDeath
DKEE	HeroKeeperOfTheGroveDeath

DKNI	KnightDeath
DKOB	KoboldDeath
DKOD	KodoBeastDeath
DLIC	HeroLichDeath
DLOC	LocustDeath
DLVR	LeverDeath
DMAG	MagnataurDeath
DMAK	MakruraDeath
DMAM	MammothDeath
DMGS	MurgulDeathSwim
DMGT	MountainGiantDeath
DMKG	HeroMountainKingDeath
DMLF	MalfurionDeath
DMOO	HeroMoonPriestessDeath
DMOR	MortarTeamDeath
DMTW	MeatWagonDeath
DMUR	MurlocDeath
DNBL	NagaBuildingDeath
DNDR	NetherDragonDeath
DNDS	HumanDissipate
DNEC	NecromancerDeath
DMYR	NagaMyrmidonDeath
DMYS	NagaMyrmidonDeathSwim
DNSR	NagaSirenDeath
DNSS	NagaSirenDeathSwim
DNSW	NagaSeaWitchDeath
DNWS	NagaSeaWitchDeathSwim
DOBS	ObsidianStatueDeath
DOGR	OgreDeath
DOLB	DeathOrcLargeBuilding
DOLS	DeathOrcSmallBuilding
DORW	OrcWarlockDeath
DOWB	OwlbearDeath
DOWL	SnowOwlDeath
DPAL	HeroPaladinDeath
DPMB	PandarenBrewmasterDeath
DPEN	PenguinDeath
DPEO	PeonDeath
DPES	PeasantDeath
DPHX	PhoenixDeath
DPIG	PigDeath
DPIT	CryptFiendDeath
DPLD	PitlordDeath
DPRS	PriestDeath
DQBS	QuillBeastDeath
DRAI	RaiderDeath
DRAN	ArcherDeath
DRAT	RatDeath
DREV	RevenantDeath
DRKG	RockGolemDeath
DRKW	RockWallDeath
DRHG	RiddenHippogryphDeath
DRIF	RiflemanDeath
DSAT	SatyrDeath
DSCB	ScarabDeath
DSEL	SealDeath
DSEN	SentinelDeath
DSGT	SeaGiantDeath
DSGW	SeaGiantSwimDeath
DSHD	ShadeDeath
DSHH	HeroShadowHunterDeath
DSHM	ShamanDeath
DSHP	SheepDeath
DSHW	SheepDeathSwim
DSKE	SkeletonDeath
DSKK	SkinkDeath
DSLG	SludgeMonsterDeath
DSND	SnapDragonDeath
DSNS	SnapDragonDeathSwim
DSOR	SorceressDeath

DSPB	SpellBreakerDeath
DSPC	SpiderCrabDeath
DSPD	SpiderDeath
DSPL	SplatDeath
DSPV	SpiritOfVengeanceDeath
DSTT	SteamTankDeath
DTAU	TaurenDeath
DTCH	HeroTaurenChieftainDeath
DTRW	TreeWallDeath
DTUS	TuskarrDeath
DUAB	UndeadAirBargeDeath
DUBC	DeathUndeadBuildingCancel
DUDS	UndeadDissipate
DULB	DeathUndeadLargeBuilding
DULS	DeathUndeadSmallBuilding
DUNB	UnbrokenDeath
DVLC	VillagerChildDeath
DVLM	VillagerManDeath
DVLW	VillagerWomanDeath
DVNG	VengeanceDeath
DVUL	VultureDeath
DWAR	WarlockDeath
DWAT	WaterElementalDeath
DWCD	WyvernCageDeath
DWDS	WingedSerpentDeath
DWEN	WendigoDeath
DWIT	WitchDoctorDeath
DWLD	WarlordDeath
DWLF	WolfDeath
DWRD	HeroWardenDeath
DWRE	WarEagleDeath
DWSP	WispDeath
DWYV	WyvernRiderDeath
DZOM	ZombieDeath
KANG	AncestralGuardianAttack1
KAOE	AncientOfTheEarthAttack1
KAOM	AncientOfTheMoonAttack1
KAW1	AncientOfTheWildAttack1
KAW2	AncientOfTheWildAttack2
KANP	AncientProtectorMissileAttack
KAP1	AncientProtectorMeleeAttack1
KAP2	AncientProtectorMeleeAttack2
KBAL	BalrogAttack1
KAZB	AzureDragonAttack1
KBLL	BallistaAttack
KBLB	BlackDragonAttack1
KBRB	BronzeDragonAttack1
KBST	BeastmasterAttack
KGRB	GreenDragonAttack1
KRDB	RedDragonAttack1
KBM1	HeroBladeMasterAttack1
KBM2	HeroBladeMasterAttack2
KCAN	CannonTowerAttack
KCAT	CatapultAttack1
KCL1	CryptLordAttack1
KCL2	CryptLordAttack2
KDH1	HeroDemonHunterAttack1
KDH2	HeroDemonHunterAttack2
KDK1	HeroDeathKnightAttack1
KFAR	HeroFarSeerAttack1
KFRB	FrostWyrmAttack1
KGUA	GuardTowerAttack
KGYR	GyrocopterAttack
KIN1	InfernalAttack1

KIN2	InfernalAttack2
KINJ	InfernalJuggernaughtAttack
KINM	InfernalMachineAttack
KIRG	IronGolemAttack1
KLIC	HeroLichAttack1
KMKG	HeroMountainKingAttack1
KMT1	MortarTeamAttack1
KMT2	MortarTeamAttack2
KMTW	MeatWagonAttack1
KPB1	BrewmasterAttack1
KPB2	BrewmasterAttack2
KPL1	HeroPaladinAttack1
KPL2	HeroPaladinAttack2
KPD1	PitLordAttack1
KPD2	PitLordAttack2
KPD3	PitLordAttack3
KPS1	PitLordAttackSlam1
KPS2	PitLordAttackSlam2
KRG1	RockGolemAttack1
KRG2	RockGolemAttack2
KRIF	RiflemanAttack1
KRN1	HeroRangerAttack1
KRN2	HeroRangerAttack2
KSTT	SteamTankAttack
KTC1	HeroTaurenChieftainAttack1
KTC2	HeroTaurenChieftainAttack2
KTOL	TreeOfLifeAttack1
KWAR	WardenAttack
MABS	AbsorbManaLaunch
MANG	AncestralGuardianMissileHit
MANL	AncestralGuardianMissileLaunch
MANP	AncientProtectorMissileHit
MAPL	AncientProtectorMissileLaunch
MARL	ArrowLaunch
MARR	ArrowHit
MAXE	AxeMissileHit
MAXL	AxeMissileLaunch
MBAL	BallistaMissileHit
MBAN	BansheeMissileHit
MBHT	BoatMissileHit
MBHL	BoatMissileLaunch
MBML	BloodMageMissileLaunch
MBNL	BansheeMissileLaunch
MBLT	Bolt
MBRH	BristleBackMissileHit
MBRL	BristleBackMissileLaunch
MBSL	PriestMissileLaunch
MBSH	PriestMissileHit
MCAH	ChimaeraAcidHit
MCAL	ChimaeraAcidLaunch
MCAN	CannonTowerMissile
MCAT	Catapult
MCDA	ColdArrow
MCRH	CryptFiendMissileHit
MCRL	CryptFiendMissileLaunch
MDCL	DeathCoilMissile
MDEM	DemonHunterMissileHit
MDLL	DestroyerMissileLaunch
MDML	DemonHunterMissileLaunch
MDOC	WitchDoctorMissileLaunch
MDOH	WitchDoctorMissileHit
MDRY	DryadMissile
MDTL	DruidOfTheTalonMissileLaunch
MDTH	DruidOfTheTalonMissileHit

MDVM	DevourMagicLaunch
MFAH	FrostArrowHit
MFAL	FrostArrowLaunch
MFAR	FarseerMissile
MFBL	FrostBoltLaunch
MFBH	FrostBoltHit
MFDL	FaerieDragonLaunch
MFKH	FanOfKnivesHit
MFLA	SearingArrowHit
MFLL	SearingArrowLaunch
MFRB	Fireball
MFRL	FireballLaunch
MGML	GryphonRiderMissileLaunch
MGRH	GargoyleMissileHit
MGRL	GargoyleMissileLaunch
MGUA	GuardTowerMissileHit
MHAR	HarpyMissileHit
MHRL	HarpyMissileLaunch
MHNL	HunterMissileLaunch
MHUN	HunterMissileHit
MKGL	KeeperOfTheGroveMissileLaunch
MKGH	KeeperOfTheGroveMissileHit
MKML	NecromancerMissileLaunch
MKMH	NecromancerMissileHit
MLIC	LichMissile
MLSL	BansheeMissileLaunch
MLSH	BansheeMissileHit
MMEA	MeatWagonMissileHit
MMTI	Mortar
MNCH	NecromancerMissileHit
MNCL	NecromancerMissileLaunch
MPAH	PoisonArrowHit
MPML	PriestMissileLaunch
MPMH	PriestMissileHit
MPXL	PhoenixMissileLaunch
MRAN	RangerMissile
MRIF	Rifle
MSBL	PriestMissileLaunch
MSBH	PriestMissileHit
MSEH	SentinelMissileHit
MSEL	SentinelMissileLaunch
MSHD	ShadowHunterMissileLaunch
MSHH	ShadowHunterMissileHit
MSMH	SorceressMissileHit
MSML	SorceressMissileLaunch
MSNL	SnapDragonMissileLaunch
MSPR	Spear
MSVL	GargoyleMissileLaunch
MSVH	GargoyleMissileHit
MTBL	TrollBatriderMissileLaunch
MWAT	WaterElementalMissile
MWEB	WebMissileLaunch
MWIN	DragonHawkMissileHit
MWNL	DragonHawkMissileLaunch
MWYV	WyvernSpearMissile
MZIG	ZigguratMissileLaunch
MZGH	ZigguratMissileHit
MZFL	ZigguratFrostMissileLaunch
MZFH	ZigguratFrostMissileHit
GSMN	ExpansionGlueMonster
AIFT	FinalCinematic

Appendix F - Spawn Models

UEGG	Objects\Spawnmodels\Undead\CryptFiendEggsack\CryptFiendEggsack.mdl
GCBL	Objects\Spawnmodels\Undead\GargoyleCrumble\GargoyleCrumble.mdl
UDIS	Objects\Spawnmodels\Undead\UndeadDissipate\UndeadDissipate.mdl
EDIS	Objects\Spawnmodels\NightElf\NightelfDissipate\NightElfDissipate.mdl
DDIS	Objects\Spawnmodels\Demon\DemonDissipate\DemonDissipate.mdl
ODIS	Objects\Spawnmodels\Orc\OrcDissipate\OrcDissipate.mdl
HDIS	Objects\Spawnmodels\Human\HumanDissipate\HumanDissipate.mdl
HBS0	Objects\Spawnmodels\Human\HumanBlood\HumanBloodSmall0.mdl
HBS1	Objects\Spawnmodels\Human\HumanBlood\HumanBloodSmall1.mdl
HBL0	Objects\Spawnmodels\Human\HumanBlood\HumanBloodLarge0.mdl
HBL1	Objects\Spawnmodels\Human\HumanBlood\HumanBloodLarge1.mdl
EENT	Objects\Spawnmodels\NightElf\EntBirthTarget\EntBirthTarget.mdl
DNAM	Objects\Spawnmodels\NightElf\NEDeathMedium\NEDeath.mdl
DNAS	Objects\Spawnmodels\NightElf\NEDeathSmall\NEDeathSmall.mdl
DUME	Objects\Spawnmodels\Undead\UDeathMedium\UDeath.mdl
DUSM	Objects\Spawnmodels\Undead\UDeathSmall\UDeathSmall.mdl
INFR	Objects\Spawnmodels\Demon\InfernalMeteor\InfernalMeteor.mdl
INFL	Objects\Spawnmodels\Demon\InfernalMeteor\InfernalMeteor2.mdl
INFU	Objects\Spawnmodels\Demon\InfernalMeteor\InfernalMeteor3.mdl
HBF0	Objects\Spawnmodels\Human\HumanBlood\HumanBloodFootman.mdl
HBK0	Objects\Spawnmodels\Human\HumanBlood\HumanBloodKnight.mdl
HBM0	Objects\Spawnmodels\Human\HumanBlood\HumanBloodMortarTeam.mdl
HBP0	Objects\Spawnmodels\Human\HumanBlood\HumanBloodPeasant.mdl
HBPR	Objects\Spawnmodels\Human\HumanBlood\HumanBloodPriest.mdl
HBR0	Objects\Spawnmodels\Human\HumanBlood\HumanBloodRifleman.mdl
HBSR	Objects\Spawnmodels\Human\HumanBlood\HumanBloodSorceress.mdl
HBNE	Objects\Spawnmodels\Undead\UndeadBlood\UndeadBloodNecromancer.mdl
NBVW	Objects\Spawnmodels\Other\NPCBlood\NpcBloodVillagerWoman.mdl
OBHE	Objects\Spawnmodels\Orc\Orcblood\OrcBloodHeadhunter.mdl
OBHS	Objects\Spawnmodels\Orc\Orcblood\OrcBloodHellScream.mdl
OBFS	Objects\Spawnmodels\Orc\Orcblood\OrcBloodHeroFarSeer.mdl
OBTC	Objects\Spawnmodels\Orc\Orcblood\OrcBloodHeroTaurenChieftain.mdl
OBKB	Objects\Spawnmodels\Orc\Orcblood\OrcBloodKotoBeast.mdl
OBWD	Objects\Spawnmodels\Orc\Orcblood\OrcBloodWitchDoctor.mdl
OBWR	Objects\Spawnmodels\Orc\Orcblood\OrcBloodWolfrider.mdl
OBWY	Objects\Spawnmodels\Orc\Orcblood\OrdBloodWyvernRider.mdl
OBWV	Objects\Spawnmodels\Orc\Orcblood\OrcBloodRiderlessWyvernRider.mdl
OBT0	Objects\Spawnmodels\Orc\Orcblood\OrcBloodTauren.mdl
OBG0	Objects\Spawnmodels\Orc\Orcblood\OrcBloodGrunt.mdl
OBP0	Objects\Spawnmodels\Orc\Orcblood\OrcBloodPeon.mdl
OKBP	Objects\Spawnmodels\Orc\KodoBeastPuke\KodoBeastPuke.mdl
UBGA	Objects\Spawnmodels\Undead\UndeadBlood\UndeadBloodGargoyle.mdl
UBGH	Objects\Spawnmodels\Undead\UndeadBlood\UndeadBloodGhoul.mdl
UBAB	Objects\Spawnmodels\Undead\UndeadBlood\UndeadBloodAbomination.mdl
UBAC	Objects\Spawnmodels\Undead\UndeadBlood\UndeadBloodAcolyte.mdl
DBCR	Objects\Spawnmodels\Undead\UndeadBlood\UndeadBloodCryptFiend.mdl
NBAR	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodArcher.mdl
NBDC	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodDruidoftheClaw.mdl
NBDT	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodDruidoftheTalon.mdl
NBDR	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodDryad.mdl
NBHU	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodHuntress.mdl
NBDB	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodDruidBear.mdl
NBDA	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodDruidRaven.mdl
NBDH	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodHeroDemonHunter.mdl
NBKG	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodHeroKeeperoftheGrove.mdl
NBMP	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodHeroMoonPriestess.mdl
NBCH	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodChimaera.mdl
NBHG	Objects\Spawnmodels\NightElf\NightElfBlood\NightElfBloodHippogryph.mdl
DBPT	Objects\Spawnmodels\Demon\DemonBlood\DemonBloodPitlord.mdl
DNBL	Objects\Spawnmodels\Other\NeutralBuildingExplosion\NeutralBuildingExplosion.mdl
CLID	Objects\Spawnmodels\Undead\ImpaleTargetDust\ImpaleTargetDust.mdl
HFSS	Objects\Spawnmodels\Human\SmallFlameSpawn\SmallFlameSpawn.mdl
UBSC	Objects\Spawnmodels\Undead\UndeadBlood\ObsidianStatueCrumble.mdl
UBCC	Objects\Spawnmodels\Undead\UndeadBlood\ObsidianStatueCrumble2.mdl
HBBM	Objects\Spawnmodels\Human\HumanBlood\HeroBloodElfBlood.mdl
HBSB	Objects\Spawnmodels\Human\HumanBlood\BloodElfSpellThiefBlood.mdl
NBMF	Objects\Spawnmodels\NightElf\NightElfBlood\MALFurion_Blood.mdl

OBBT	Objects\Spawnmodels\Orc\Orcblood\BattrollBlood.mdl
OBSH	Objects\Spawnmodels\Orc\Orcblood\HeroShadowHunterBlood.mdl
DBPB	Objects\Spawnmodels\Other\PandarenBrewmasterBlood\PandarenBrewmasterBlood.mdl
DBBM	Objects\Spawnmodels\Other\BeastmasterBlood\BeastmasterBlood.mdl
PEFI	Abilities\Spells\Other\ImmolationRed\ImmolationREDTarget.mdl
DNBD	Objects\Spawnmodels\Naga\NagaDeath\NagaDeath.mdl
FTSO	Objects\Spawnmodels\Other\FlameThrower\FlameThrowerSpawnObj.mdl
TOBO	Objects\Spawnmodels\Other\ToonBoom\ToonBoom.mdl
CBAL	Objects\Spawnmodels\Critters\Albatross\CritterBloodAlbatross.mdl
IFP0	Objects\Spawnmodels\Other\IllidanFootprint\IllidanSpawnFootPrint0.mdl
IFP1	Objects\Spawnmodels\Other\IllidanFootprint\IllidanSpawnFootPrint1.mdl
IFPW	Objects\Spawnmodels\Other\IllidanFootprint\IllidanWaterSpawnFootPrint.mdl
HBCE	Objects\Spawnmodels\Other\HumanBloodCinematicEffect\HumanBloodCinematicEffect.mdl
OBCE	Objects\Spawnmodels\Other\OrcBloodCinematicEffect\OrcBloodCinematicEffect.mdl
FRBS	Objects\Spawnmodels\Human\FragmentationShards\FragBoomSpawn.mdl
PBSX	Objects\Spawnmodels\Other\PandarenBrewmasterExplosionUltimate\PandarenBrewmasterExplosionUltimate.
GDCR	UI\Feedback\GoldCredit\GoldCredit.mdl
NBWS	Objects\Spawnmodels\Naga\NagaBlood\NagaBloodWindserpent.mdl

YOU SHOULD CAREFULLY READ THE FOLLOWING END USER LICENSE AGREEMENT BEFORE INSTALLING THIS SOFTWARE PROGRAM. BY INSTALLING, COPYING, OR OTHERWISE USING THE SOFTWARE PROGRAM, YOU AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, PROMPTLY DELETE ANY OF THE FILES THAT YOU HAVE DOWNLOADED RELATED TO THIS SOFTWARE PROGRAM.

This software program (the "Program"), any printed materials, any online or electronic documentation, and any and all copies, derivative works of such software program and materials created with the Program, are the copyrighted work of Blizzard Entertainment, a division of Davidson & Associates, Inc. and/or its licensors, including, but not necessarily limited to, Autodesk, Inc. All use of the Program is governed by the terms of the End User License Agreement which is provided below ("License Agreement"). The Program is solely for use by end users according to the terms of the License Agreement. Any use, reproduction or redistribution of the Program not in accordance with the terms of the License Agreement is expressly prohibited.

END USER LICENSE AGREEMENT

1. Limited Use License. Blizzard Entertainment ("Blizzard") hereby grants, and by downloading and/or installing the Program you thereby accept, a limited, non-exclusive license and right to download and install and use one (1) copy of the Program for your use on either a home or portable computer. The Program contains software which is utilized as a plug-in for Autodesk's 3Ds Max software product to create certain art files that can produce standalone graphic illustrations, as well as operate in conjunction with the Warcraft® III 'Campaign Editor' (the "Editor") to allow you to create custom levels or other materials for your personal use in connection with the Program ("New Materials"). Use of 3Ds Max is governed by the Autodesk, Inc. license agreement. All use of the New Materials is subject to this License Agreement. In addition, the Program can be utilized with the Editor to create New Materials that can be utilized in conjunction with Warcraft III to play over Blizzard Entertainment's online game network, Battle.net. Use of New Materials on Battle.net is subject to your acceptance of the Battle.net Terms of Use Agreement. Blizzard Entertainment reserves the right to update, modify or change the Battle.net Terms of Use Agreement at any time. All terms and conditions referenced therein are included herein, and in the event of a conflict in terms between the Warcraft III End User License Agreement and this License Agreement, the terms and conditions of the Warcraft III End User License Agreement shall prevail.

2. Ownership. All title, ownership rights and intellectual property rights in and to the Program and any and all copies thereof, and any New Materials (including but not limited to any titles, computer code, themes, objects, characters, character names, stories, dialog, catch phrases, locations, concepts, artwork, animations, sounds, musical compositions, audio-visual effects, methods of operation, moral rights, any related documentation, and "applets" incorporated into the Program) are owned by Blizzard Entertainment or its licensors. The Program is protected by the copyright laws of the United States, international copyright treaties and conventions and other laws. All rights are reserved. The Program contains certain licensed materials, and Blizzard's licensors may protect their rights in the event of any violation of this Agreement.

3. Responsibilities of End User.

A. Subject to the Grant of License hereinabove, you may not, in whole or in part, copy, photocopy, reproduce, translate, reverse engineer, derive source code from, modify, disassemble, decompile, create derivative works based on, or remove any proprietary notices or labels on the Program without the prior consent, in writing, of Blizzard.

B. The Program is licensed to you as a single product. Its component parts may not be separated for use on more than one computer.

C. You are entitled to use the Program for your own use, but you are not entitled to:

(i) sell, grant a security interest in or transfer reproductions of the Program, or any New Materials, to other parties in any way, nor to rent, lease or license the Program or any New Materials to others without the prior written consent of Blizzard;

(ii) exploit the Program, any of its parts, or any New Materials for any commercial purpose including, but not limited to, use at a cyber cafe, computer gaming center or any other location-based site (Blizzard may offer a separate Site License Agreement to permit you to make the Program available for commercial use; contact Blizzard for details); or

(iii) use or allow third parties to use the Program, Editor and/or the New Materials created thereby for commercial purposes including, but not limited to, distribution of New Materials on a standalone basis or packaged with other software or hardware through any and all distribution channels, including, but not limited to, retail sales and online electronic distribution, without the express written consent of Blizzard.

4. Program Transfer. You may permanently transfer all of your rights under this License Agreement, provided the recipient agrees to the terms of this License Agreement and you agree to remove the Program and any New Materials from your home or portable computer.

5. Termination. This License Agreement is effective until terminated. You may terminate the License Agreement at any time by destroying the Program and any New Materials. Blizzard may, at its discretion, terminate this License Agreement in the event that you fail to comply with the terms and conditions contained herein. In such event, you must immediately destroy the Program and any New Materials.

6. Export Controls. The Program may not be re-exported, downloaded or otherwise exported into (or to a national or resident of) any country to which the U.S. has embargoed goods, or to anyone on the U.S. Treasury Department's list of Specially Designated Nationals or the U.S. Commerce Department's Table of Denial Orders. By installing the Program, you are agreeing to the foregoing, and you are representing and warranting that you are not located in, under the control of, or a national or resident of any such country or on any such list.

7. Disclaimer of Any Limited Warranty. Blizzard expressly disclaims any warranty for the Program and Manual(s). The Program and Manual(s) are provided "as is" without warranty of any kind, either express or implied, including, without limitation, the implied warranties of merchantability, fitness for a particular purpose, or noninfringement. The entire risk arising out of use or performance of the Program and Manual(s) remains with you, the User. Some states do not allow the exclusion or limitation of implied warranties or liability for incidental damages, so the above limitations may not apply to you.

8. Limitation of Liability. NEITHER BLIZZARD NOR ITS PARENT, SUBSIDIARIES, AFFILIATES OR LICENSORS SHALL BE LIABLE IN ANY WAY FOR LOSS OR DAMAGE OF ANY KIND RESULTING FROM THE USE OF THE PROGRAM, THE EDITOR OR BLIZZARD ENTERTAINMENT'S ONLINE GAME NETWORK, BATTLE.NET, INCLUDING, BUT NOT LIMITED TO, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES. ANY WARRANTY AGAINST INFRINGEMENT THAT MAY BE PROVIDED IN SECTION 2-312(3) OF THE UNIFORM COMMERCIAL CODE AND/OR IN ANY OTHER COMPARABLE STATE STATUTE IS EXPRESSLY DISCLAIMED. FURTHER, BLIZZARD ENTERTAINMENT SHALL NOT BE LIABLE IN ANY WAY FOR THE LOSS OR DAMAGE TO PLAYER CHARACTERS, ACCOUNTS, STATISTICS OR USER PROFILE INFORMATION STORED ON BATTLE.NET. BLIZZARD ENTERTAINMENT SHALL NOT BE RESPONSIBLE FOR ANY INTERRUPTIONS OF SERVICE ON BATTLE.NET, INCLUDING, BUT NOT LIMITED TO, ISP DISRUPTIONS, SOFTWARE OR HARDWARE FAILURES OR ANY OTHER EVENT WHICH MAY RESULT IN A LOSS OF DATA OR DISRUPTION OF SERVICE. Some states do not allow the exclusion or limitation of incidental or consequential damages, or allow limitations on how long an implied warranty lasts, so the above limitations may not apply.

9. Equitable Remedies. You hereby agree that Blizzard would be irreparably damaged if the terms of this License Agreement were not specifically enforced, and therefore you agree that Blizzard shall be entitled, without bond, other security, or proof of damages, to appropriate equitable remedies with respect to breaches of this License Agreement, in addition to such other remedies as Blizzard may otherwise have available to it under applicable laws. In the event any litigation is brought by either party in connection with this License Agreement, the prevailing party in such litigation shall be entitled to recover from the other party all the costs, attorneys' fees and other expenses incurred by such prevailing party in the litigation.

10. Limitations on License. Nothing in this License Agreement shall preclude you from making or authorizing the making of another copy or adaptation of the Program provided, however, that (1) such new copy or adaptation is created as an essential step in your utilization of the Program in accordance with the terms of this License Agreement and for NO OTHER PURPOSE; or (2) such new copy or adaptation is for archival purposes ONLY, and all archival copies are destroyed in the event of your Transfer of the Program, the Termination of this Agreement or other circumstances under which your continued use of the Program ceases to be rightful.

11. Miscellaneous. This License Agreement shall be deemed to have been made and executed in the State of California, and any dispute arising hereunder shall be resolved in accordance with the law of California. You agree that any claim asserted in any legal proceeding by one of the parties against the other shall be commenced and maintained in any state or federal court located in the State of California, County of Los Angeles, having subject matter jurisdiction with respect to the dispute between the parties. This License Agreement may be amended, altered or modified only by an instrument in writing, specifying such amendment, alteration or modification, executed by both parties. In the event that any provision of this License Agreement shall be held by a court or other tribunal of competent jurisdiction to be unenforceable, such provision will be enforced to the maximum extent permissible, and the remaining portions of this License Agreement shall remain in full force and effect. This License Agreement constitutes and contains the entire agreement between the parties with respect to the subject matter hereof and supersedes any prior oral or written agreements.