

Exporting Animated Creatures

Welcome to the guide to exporting animated creatures. The purpose of this guide is to give instructions on how to export a creature's meshes, skeletons and animations. This guide assumes you have already built, rigged and animated a creature in 3D Studio Max and are now ready to export.

TABLE OF CONTENTS

Terminology and General Information	2
Exporting Skeletons	2
Exporting a Body Skeleton (.gr2 file).....	2
Exporting a Head Skeleton (.gr2 file)	3
Exporting Meshes.....	3
Naming Meshes	3
Exporting a Body Mesh (.mdb file)	4
Exporting a Head Mesh (.mdb file).....	5
Exporting Animations	5
Naming Animations.....	5
Exporting Animations (.gr2 file).....	5
Naming Information	6
Creature type.....	6
Creature name	6
Armor Types	6
Weapon Stances.....	6
Skeleton Name.....	6
Animation Name.....	6
Troubleshooting.....	7
First areas to check.....	7
Bone Limit	7
Fixing Vertexes that are causing errors with Expotron	7
Sometimes a creature won't play its idle cycle animation.....	7

TERMINOLOGY AND GENERAL INFORMATION

Important Note on Naming Structure:

This document will go into detail about naming structures and conventions. Objects surrounded by angle brackets should be replaced by the intended word. For example <creature name> would be replaced by the word "zombie" (without quotes) for a zombie creature.

Mesh: A 3D Shape made up of polygons.

Skin: A mesh that has been modified to follow a skeleton.

Skeleton: A term referring to the hierarchy of bones (or joints) inside a creature. These bones aid creature animation, similar to how bones in the human body cause skin to move.

Skeleton Root Node: The point at which the skeleton hierarchy starts. (the highest point in the skeleton chain of command) This node is typically the pelvis bone on a creature. For example, if the root (pelvis) is moved the entire skeleton moves with it.

Attachment Points (APs): A Dummy object set up 3d Studio Max. The Nwn2 engine uses APs for attaching one mesh to a fixed point another mesh. For example, an AP on a creature's arm would allow for a shield mesh to attach to that spot the arm and follow its movements.

Collision Spheres: A collection of primitive 3D sphere objects set up in 3D studio max. NWN2 utilizes a multi-layered collision system, made up of complex collision objects within simple collision objects – making for finer collision and selection detection when necessary. Collision spheres, the lowest creature level detection (allowing for the finest level of selection/collision) are small collision objects attached to various points of a creature's skeleton. These spheres are located throughout a creature and animate with the skeleton. One function of these spheres is used by the NWN2 engine to determine if the player is trying click on a creature. For example a collision sphere on a creature's head would allow for that creature's head to be clickable by the player.

.gr2 file: Granny file format. This contains creature skeleton and animation information.

.mdb file: Neverwinter Nights 2 mesh file format. This contains mesh information.

EXPORTING SKELETONS

Exporting a Body Skeleton (.gr2 file)

Exporting the skeleton is your first step to exporting a creature. You will need to do this before you can export the creature's meshes. Keep in mind that the root of the skeleton (typically the pelvis) must be named in conjunction with your skeleton file name. Check out the "Naming Information" section for further details.

Naming the skeleton root node

Name the root node of the skeleton in this format: <creature type>_<skeleton name>
Examples include naming the root node: "P_HHM" or "c_dogleg"

Naming the body skeleton file

File naming structure for a body skeleton: <creature type>_<skeleton name>_skel.gr2
Examples include: "P_HHM_skel" or "c_dogleg_skel"

Exporting steps

- 1) Make sure your creature is at world coordinate 0,0,0 and it is returned to its original bind pose (neutral pose used in skinning process).
- 2) Select the creature's full body skeleton hierarchy and all APs associated with that skeleton. (wings, capes and other separate skeleton hierarchies are exported separately)
- 3) *optional* Create a selection set for that skeleton and name it accordingly (ex. "ExportBodySkel")
- 4) File > Export Selected and choose: save as type "Granny Run-time (.gr2)"
- 5) Name it <creaturetype>_<creaturename>_skel.gr2 (ex. P_hhm_skel)
- 6) At the export window only export "models"
- 7) Make sure "include reference in export" is checked
- 8) Make sure "move to origin" is unchecked
- 9) Double Click "Models" to see subparts
- 10) Double Click creature's name and insure all individual parts have the same settings.
- 11) Click "Export"

Exporting a Head Skeleton (.gr2 file)

Not all creatures will have a head skeleton. Head skeletons are used when there is more than one head mesh variant for a creature. This allows for variations in the facial bone positions for each head mesh.

Important head skeleton considerations

The name of the skeleton file must correspond with the name of the head mesh variant number. The head skeleton MUST have the same amount of bones as the body skeleton and the bones MUST be named the same as bones in the body skeleton.

Naming the head skeleton file

File naming structure is as follows: <creature type>_<skeleton name>_skel<head mesh variation number>.gr2
Examples include: "P_HHM_skel01.gr2" or "c_dogleg_skel01.gr2"

Important: The number in the head skeleton file name must correspond with the number in the head mesh file name. For example: "P_HHM_head05", must use head skeleton "P_HHM_skel05"

Exporting

Exporting a head skeleton is the same process as exporting a body skeleton, the only difference is the file name structure.

EXPORTING MESHES

Naming Meshes

Below is what you should name your meshes inside 3D Studio Max. The name of the mesh file (.mdb) you export must be the same as the name of the mesh in 3D Studio Max. Check out the "Naming Information" section for further details.

Naming the body mesh

Body mesh naming structure is as follows: <creature type>_<creature name>_<armor class>_body<body variant number>
Examples include: "C_Lizman_CL_Body01 " or "P_HHM_LE_Body06 "

Naming the head mesh

Head mesh naming structure is as follows: <creature type>_<creature name>_head<head variant number>
Examples include: " C_Orc_Head01" or "P_HHM_Head05"

Important: The number in head mesh name must correspond with the number in the head skeleton file name. For example: "P_HHM_head05", must use head skeleton "P_HHM_skel05"

Naming the eye mesh

Eye mesh naming structure is as follows: <creature type>_<creature name>_eye<head variant number>
Examples include: " C_Orc_Eye01" or "P_HHM_Eye05"

Naming the facial hair mesh

Facial mesh naming structure is as follows: <creature type>_<creature name>_head<head variant number>_fhair01
Examples include: " P_HHM_Head05_fhair01"

Naming LoD meshes

LoD (level of detail) meshes can be added simply by naming the LOD mesh the same as the original mesh and adding _L01, L02, ect.. An example: "P_HHM_LE_Body06_L01"
Note This is L (the number zero) 2, not L (the letter O) 2

Exporting a Body Mesh (.mdb file)

After you have exported the creature's skeletons and named your meshes according to the instructions above, you can begin exporting.

Important: Turn off any direct-x plugins like "metal bump 9" before exporting a mesh. These plugins can interfere with how Expotron reads the texture information for the mesh

- 1) Make sure you have the corresponding body skeleton (.gr2) file in the folder you are exporting the body mesh (.mdb) file to.
- 2) Make sure your creature is at world coordinate 0,0,0 and it is returned to its original bind pose (neutral pose used in skinning process).
- 3) Select Body Mesh, Body Mesh LoDs and Collision spheres.
- 4) *optional* Create a selection set for those meshes and name it accordingly (ex. "ExportBodyMesh")
- 5) Go to Tools > Expotron utility.
- 6) Select "No Transparency" or "Transparency Mask" depending on if an alpha map is used.
- 7) Select "Character (skin)".
- 8) Under "Skeleton Name" enter the file name of the skeleton file the mesh will be using. (ex. P_HHM_skel.gr2)
- 9) Select mesh, LOD's and Connected collision spheres
- 10) File>Export Selected and choose: save as type Neverwinter Nights 2 (".mdb")
- 11) Name the mesh file <creature type>_<creature name>_<armor class>_body<body variant number>.mdb (ex. P_HHM_LE_Body06.mdb)
- 12) Make sure all boxes are checked in the pop up window.

Exporting a Head Mesh (.mdb file)

- 1) Make sure you have the corresponding head skeleton (.gr2) file in the folder you are exporting the head mesh (.mdb) file to.
- 2) Make sure your creature is at world coordinate 0,0,0 and it is returned to its original bind pose (neutral pose used in skinning process).
- 3) Select the head mesh, eye mesh, facial hair mesh, related LoDs and collision sphere.
- 4) *optional* Create a selection set for those meshes and name it accordingly (ex. "ExportHeadMesh")
- 5) Go to tools > expotron
- 6) Select "No Transparency" or "Transparency Mask" depending on if an alpha map is used.
- 7) Check Head (cutscene)
- 8) Select "Character (skin)".
- 9) Under "Skeleton Name" enter the file name of the skeleton file the mesh will be using. (ex. P_HHM_skel05.gr2)
- 10) File>Export Selected and choose: save as type Neverwinter Nights 2 (".mdb")
- 11) Name as: <creature type>_<creature name>_head<head variant number>.mdb (ex. P_HHM_head05.mdb)
- 12) Make sure all boxes are checked in the pop up window.

EXPORTING ANIMATIONS

Naming Animations

Animation names play an important part of how the animations are assembled by the NWN2 engine. For example, the Gnoll, Kobold, and Lizardfolk all use the "dogleg" skeleton and therefore all share that animation set. Check out the "Naming Information" section for further details.

Naming the animation file

Naming structure is as follows: <creature type>_<skeleton name>_<weapon stance>_<animation name>.gr2

Examples include: "c_dogleg_Una_1attack01.gr2"

Exporting Animations (.gr2 file)

- 1) Go to the first frame of the animation and adjust timeline to fit desired animation.
- 2) Select the creature's entire skeleton (including any separate hierarchies, such as wings) and all APs associated with that skeleton.
- 3) *optional* Create a selection set for that selection and name it accordingly (ex. "ExportAnimations")
- 4) File > Export Selected and choose: save as type Granny Run-time (".gr2")
- 5) Name: <creature type>_<skeleton name>_<weapon stance>_<animation name>.gr2 (ex. c_zombie_una_walk.gr2)
- 6) In settings choose "animation"
 - a. "Oversampling" slide all the way to the right
 - b. Make sure "move to origin" is unchecked.
 - c. "Orientation curve" = .15
- 7) Click "Export"

Important: The NWN2 engine requires an idle cycle that is not tied to a weapon stance. Simply re-exporting the unarmed idle cycle without a weapon stance in the filename will suffice. An

example would be "P_HHM_idle.gr2" instead of "P_HHM_UNA_idle.gr2". See the "Trouble Shooting" section for more information.

NAMING INFORMATION

Creature type

The first letter in a creature's name signifies what creature type it is.

"C_" standard creature/monster

"P_" player character

"N_" non player character

Creature name

This can be any name you wish but be aware that there is a maximum file name length of 32 characters.

Armor Types

This designates what type of armor this mesh is in for appearance selection purposes in the toolset.

NK – No armor

CL – Cloth

LE – Leather

HD – Hide

SC – Scale

CH – Chain

PF – Plate, Full

Weapon Stances

This designates what stance the player is in based on what type of weapon is equipped.

1HS – One handed sword

1HSS – One handed sword and shield

C2H – Closed 2-Handed Blade

O2HT – Open 2 handed thrust (poles, staffs, spears)

O2HS – Open 2 handed slash (axes)

D2H – Dual 2 Handed

CBOW – Cross Bow

BOW – Short or Longbow

UNA – Unarmed

THRN – Sling or Dart

Skeleton Name

This can be any name you wish but be aware that there is a maximum file name length of 32 characters. All creatures using this skeleton file must have the exact same name for their skeleton hierarchy and skeleton file.

Animation Name

This can be named whatever you wish if it is going to be played using a scripted event in the game, such as in a cutscene. To be used by an action in the game, such as an unarmed attack animation, it must be named the specific name the engine expects to see. Listing all of the animation names in here would make this guide quite long. Here are the standard ones you would use for a basic creature.

walk

run
idlefidget
idleM
1attack01
1attack02
damage01
damage02
dodge01
knockdownB
proneB
standupB
taunt
death01
death02

TROUBLESHOOTING

First areas to check

- 1) Skin modifier cannot have duplicate bones.
- 2) All bones should have unique names.
- 3) Turn off any direct-x plugins like "metal bump 9" before exporting a mesh.
- 4) If a vertex is getting an error, check to see if there are more than 4 bones assigned to it.
- 5) Cinematic head skin can have no more than 3 bone weights per vertex.
- 6) Cinematic head vertexes must be weighted at least .026 to one non-face bone (f_bone)
- 7) Mesh, skeleton and animation files can only be 32 characters in filename length, not including the .mdb or .gr2 extension.

Bone Limit

There is a 54 bone limit on skeletons. This limit does not include facial bones (f_bones) used in cutscenes or attachment points (ap_dummies). Expotron will give you an error if your number of standard bones is higher than 54.

Fixing Vertexes that are causing errors with Expotron

- 1) Try using skin utilities to create a new skin mesh with vertex weights
 - a. Select both original and copy
 - b. Import skin data from mesh
 - c. Choose "match bones"
 - d. Delete copy
- 2) Try manually resetting in the weight table. (x in the "m" column) Then re-entering the correct values.

Sometimes a creature won't play its idle cycle animation

Certain aspects of the game put a creature in an idle animation that is not tied to a weapon stance, an example would be as in the toolset viewport. Simply re-exporting the unarmed idle to a new file without the unarmed designation in the file name is sufficient. For example: P_HHM_idle.gr2" instead of "P_HHM_UNA_idle.gr2".