Project Number: DMO-4012

Dis: A Neverwinter Nights Modification

A Major Qualifying Project Report
submitted to the Faculty
of the
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements for the
Degree of Bachelor of Science
By

David Corliss
Timothy Poliquin
Nicholas Urko

Date: March 6, 2009

Professor Dean M. O'Donnell, Major Advisor

Abstract

Dis is a mod of *Neverwinter Nights*. It uses an original story written for the mod. The story follows a man who is pulled into the underworld following the death of his wife. What he finds is a corrupt world from which there is no escape. The game was created using the Aurora Toolset that comes with the game *Neverwinter Nights*, as well as 3ds max, and several other game development tools.

Acknowledgements

David Corliss would like to acknowledge my parents who without I literally wouldn't be here and my partners Tim and Nick for putting up with me throughout the project.

Timothy Poliquin would like to acknowledge Maximus, for the excellent scripting tutorials, Yilmaz Kiymaz, for being our helpful play-tester, Allison, for bringing me cookies and juice during the many long days spent creating the module, and Gracie, for not doing what you normally do. Also, Mom and Dad for giving me *Neverwinter Nights* so I could play with the toolset 6 years ago, the grandparents, and anyone else who would have liked to have seen their name here but I have forgotten.

Nicholas Urko would like to acknowledge Alex Swartz for providing videos that introduced the basic art pipeline for working in *Neverwinter Nights*, Eligio Sacateca and the *Neverwinter Nights* online community for providing the *Neverwinter Nights Custom Content Guide* and a diverse set of wonderful art assets and tools to work with, and Yilmaz Kiymaz for his thorough play-testing job.

We would all like to thank Dean O'Donnell for advising us on this project and helping us across the occasional bumps in the road.

Table of Contents

Abstract **1**

Acknowledgements **2**

Introduction **5**

Building Dis **6**

The Story 6

Game Design 6

Brief Overview of the Art Pipeline 7

Making it All Work – The Aurora Toolset **9**

Area Development 10

Graphical Programming 12

NWN Script 14

Shortcomings 16

What We Would Do Differently **17**

Story 17

Art 17

Tech 18

Conclusion **20**

Appendix A **21**

Dis (Vision Statement) 21

Basic Outline 23

Detailed Outline 24

Appendix B: Dialog **27**

Appendix C: Art Walkthroughs **43**

The Art Pipeline: How to Modify/Create Content 43
The Override Folder Vs Hakpaks 43
Making Everything into a Hakpak 44
Downloading and Importing Custom Content 46
Making Custom Armor or Clothing and Recombining Hakpaks 47
Changing a Placeable's Model 53
Recoloring a Monster 56
Making New Content That Does Not Overwrite Existing Content Via .2DA Editing 58
Creatures and the Appearance.2DA File 58
Placeables and the Placeables.2DA File 65
Editing the NWN Grass 65
A Quick Overview of Tileset Editing 67
Making a Custom Door 68
Custom Lighting Effects 70
Creature Properties and Balancing Monsters 71

Appendix D: The Module **75**

Works Cited **76**

Additional Works Cited Information From the Artist **80**

Introduction

Our assigned project was to create an original module for the game *Neverwinter Nights*. To do this we had to compose an original story, learn to create new art assets and modify existing assets, and work within the Aurora Toolset. Our team for the project was composed of one game designer (who also wrote the story), a digital artist, and a programmer. As we each came in with complementing skills we were freed to embrace an ambitious project that included a story independent of the original game.

Our desire to tell a story meant that design documentation was important. In addition to the required vision statement and game design document, we also created several outlines of the story which could be distributed to the various team members. These documents allowed for the artist to locate or create the needed assets, and the programmer could begin writing code while the story was being created.

As with most creative processes, the game went through several revisions. The first version was set in modern time period but introduced surreal elements that would allow for interesting game-play. Unfortunately this version did not mesh well with the Aurora Toolset and so was scavenged to create a game more in-line with the fantasy worlds of *Neverwinter Nights*. In this second version we created a far too detailed world creating a mod that would take far too long to complete. From there we scaled back many of the areas creating the mod we are now presenting.

Building Dis

The Story

From the beginning our story was about a man who was dealing with the recent loss of his wife Betty. Originally the story called for him to experience his grief as he made his way through the city where they had lived together. However it was determined that the game engine we were using would not be suitable for this story and so we modified it. In the final version of our module the player is so overwhelmed with grief that a door to the underworld of Dis opens up to him. He enters this world uncertain as to what is happening and he is forced to enter a violent fantasy realm inside of which his wife is trapped.

The city of Dis and several of the locations, such as the Woeful Woods, are inspired by *The Inferno*, the first book in Dante Alighieri's Divine Comedy. While the stories are not in any way directly related to each other, in *The Inferno* we found an iconic depiction of the afterlife that reflected the suffering the player would be suffering from.

The world of our Dis is the image of an afterlife devoid of a real deity. An Emperor ruthlessly rules the realm, but he his petty and selfish with limited power, hiding away behind the great walls of Dis. Outside the walls, on the Scalded Plains, chaos rules as bandits terrorize the populace, and mercenaries and treasure hunters use the situation to gain wealth. Deeper into the realm, beyond the walls of Dis, wait imaginative tortures that serve as prisons for those who anger the Emperor.

Game Design

Because of the nature of our story we decided that it would be best if we provided the player with a pre-generated character. While the player can choose to use any character, we thought it best to provide them with a pre-generated character for several reasons. First, as already stated, it worked best with the story. We had a specific character in mind when writing the player's role so we found it preferable to control every aspect of his appearance and combat style. It also made balancing the game easier as having only one class to deal with meant we could make all decisions around that class and not have to build multiple paths and options needed for other classes to choose their own way.

The main bulk of the story is told through an overarching main quest. This quest begins with the player entering the realm of Dis and eventually ends when the player rescues Betty. The main quest is broken up into a series of smaller sub-objectives. These objectives serve as a means to guiding the player while highlighting the strict system of *quid pro quo* that dictates the political sphere of Dis and eventually the cruelty of the world.

In addition to the main quest there are side quests scattered throughout the world. These quests are designed to further flesh out the world, providing it with a vibrant populace and helping to better inform the player of what is happening in the world.

Brief Overview of the Art Pipeline

The art pipeline consists of a number of tools, each of which has a specific use in extracting content from *Neverwinter Nights*, creating raw files, and placing that content back in the engine. Content that is added back to the engine is added in the form of Hak Packs. Appendix C contains an indepth look at the entire art pipeline, and the process of using these programs together to get custom content into the *Neverwinter Nights* toolset.

Brief Overview of the Tools

- *Zoligato's NWN Viewer* – One of the earliest viewers for *Neverwinter Nights*. It allows you to browse all of the models, textures, and other files that come with *Neverwinter Nights*. You can extract any of these files, and then modify them to make new content. This program is mostly outdated (it was made in 2002), and can only be used to access the content for the original *Neverwinter Nights* Campaign, not any content from the expansions. This program has another feature that is still useful today. It can open .ERF files, which are *Neverwinter Nights* export files. Occasionally you will want to open these files.

- *Neverwinter Nights Explorer* - This is a more current viewer for *Neverwinter Nights* than Zoligato's. This viewer gives you access to all the content up through the second expansion. Unfortunately, it does not give you access to any of the content from the third expansion and onward.

- *Bioware's Hakpak Editor* – This is a tool that allows you to create hakpaks. Hakpaks are the most efficient method of packaging all your custom models, textures, and anything else. This tool comes standard with *Neverwinter Nights*.

- *BuildTil* – Also known as the *Neverwinter Nights* Tileset Editor. This is the official Tileset Editor used by Bioware to make *Neverwinter Nights*. It can be used to modify any existing *Neverwinter Nights* tileset, as well as any custom tileset (provided it has a .SET file). This is a very touchy program. If you use it for any length of time, you will bump into many indefinable errors, and possibly corrupt your tileset. As such, I do not recommend using this application to modify tilesets. If you really need to edit a .SET file, you would be better off opening it in Notepad and adjusting it manually. The only real use I had for this program was that it could be used as a viewer for specific tiles within a tileset. This is useful if you are looking for one or two specific tiles to modify. You can also use this program to modify the standard grass feature that shows up in some tilesets under certain graphics settings, but I strongly recommend against it.

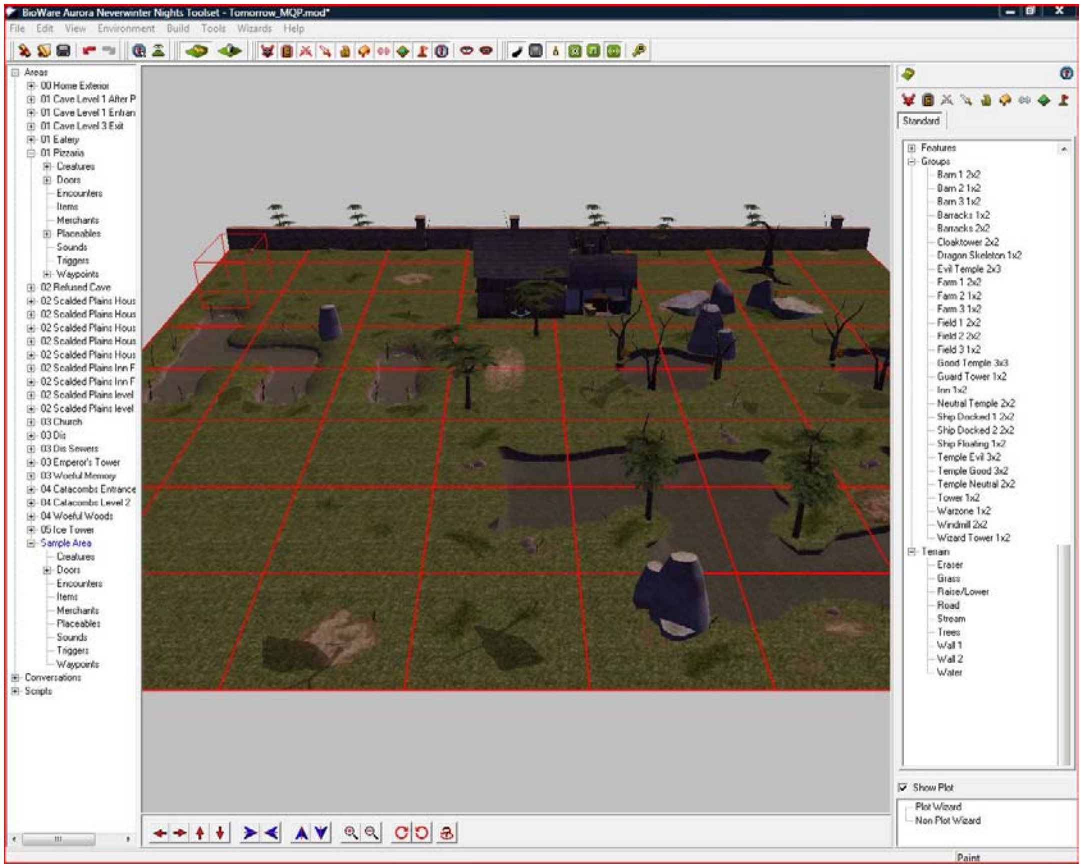
- *3D Studio Max (Version 5)* – The most compatible 3D software for use with *Neverwinter Nights*. In order to convert between *Neverwinter Nights* format and 3D Studio Max format, you need to download and install the MDL Plug-In Suite 1.1.

Making it all Work – The Aurora Toolset

Included with the Windows clients for *Neverwinter Nights* is the Aurora Toolset, a powerful application which provides players and developers with all of the assets necessary to create a traditional *Neverwinter Nights* module. By default, the toolkit includes a number of tilesets, which are used to create the basic environments, a vast array of monster and NPC creatures for developers to choose from, and even more items for users to populate areas with. All of these assets can be easily placed in an area to create a unique and engaging environment. In addition to the art, the toolset also includes over 4000 pre-made scripts for developers to use to bring life to the creatures and people placed in the world. Further still, the Aurora Toolset allows its users to create and import new assets via the use of .hak packs, and write new scripts to extend the functionality of the base game and allow for unique scenarios.

Area Development

Bioware makes creating new areas in the Aurora Toolset simple, while still allowing for diversity and flexibility. Using either the pre-packaged tilesets or custom ones created by yourself or the large online community, developers simply need to specify a name and size for an area, and they are ready to begin sculpting a new environment. By selecting the Paint Terrain button, a number of terrain types become available, depending upon the tileset used, each grouped in one of three categories. The most basic category, Terrain, is used to define the primary attributes of an area—where the walls are, what the ground looks like, where the river flows through, or where the forest begins. Then, users might move on to the Features, which allows for the placement of more specific details, like ant hills, grave sites, trees, or houses, on a single tile. Finally, Groups combine features and terrain attributes to define parts of an area over a number tiles.

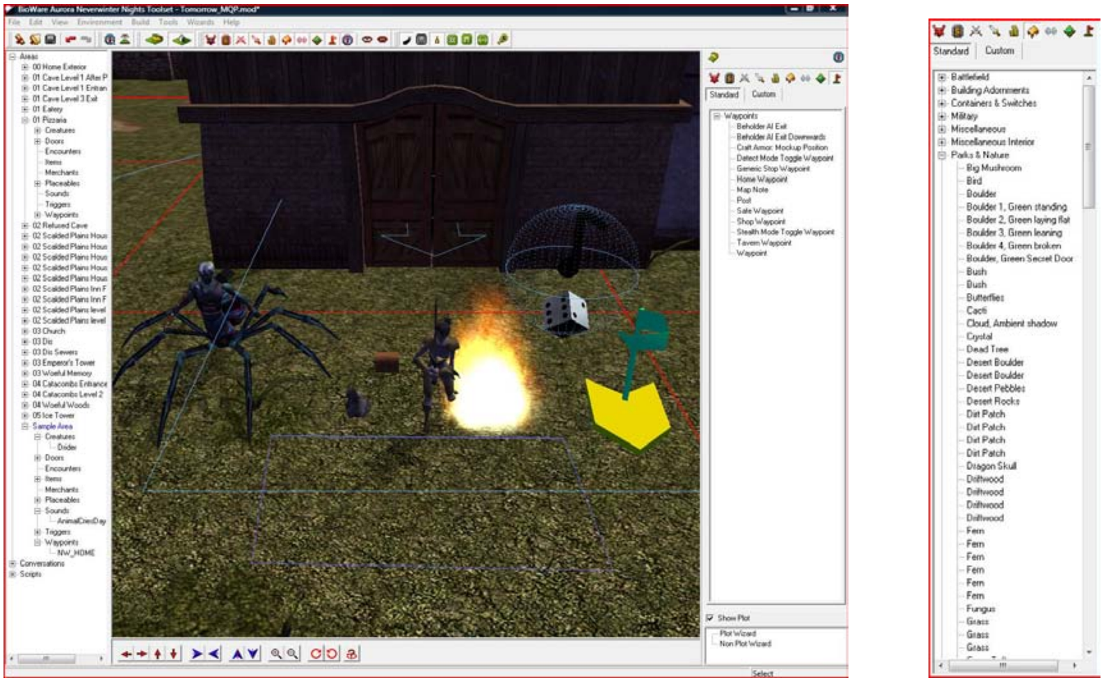


Using the Terrain Painter to create a new area

Once the area is defined, developers can begin placing things in the environment, such as monsters, people, doors, fire, and signposts. Like the tilesets, each of these objects belongs to a number of categories and sub-categories, which gradually more specifically define an object. Each of these categories is defined below.

- Creatures—the animate objects of the world. Creatures can be monsters, which the player fights to gain new items, gold, and experience, or people, whom the player speaks to in order to gather quests, learn about an area or event, or befriend.

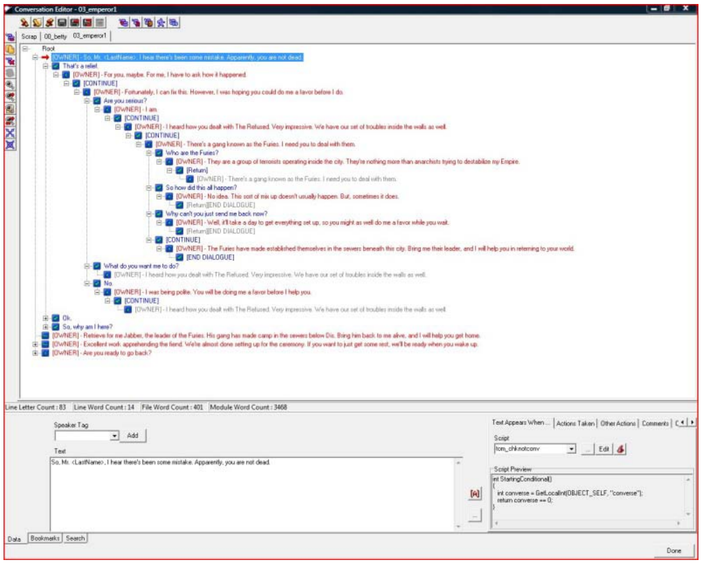
- Doors—The objects which go in door frames in order to allow the player to enter or exit an area. Doors range from traditional wooden doors to vaults.
- Encounters—the objects used to organize a particular combat scenario. Using an encounter, developers can specify the creatures to spawn in the environment, where they should spawn, and how many of them there should be. These can be specified directly, but can also be generally assigned, with allowance for variability.
- Items—the objects players can pick up in the world and use or sell to whoever is willing to buy. Items range from armor and weapons to special objects, like heads of fallen victims.
- Merchants—the people with whom the player interacts to sell or purchase new goods. Merchants have varying types of inventory and agreeability toward the player, each of which can be modified or left to a template.
- Placeables—the objects which give an environment detail and personality. Many of these objects are interactive, such as sewer grates or portals, while others remain static. However, the usability of each object is left to the developer's desire.
- Sound Effects—the objects which provide the aural component of the player's experience.
- Triggers—the objects which specify a region of action. Triggers can be used to cause the player to be swept away to a distant land, or erupt in flame. Triggers are tightly bound with scripts in order to provide unique and interesting events.
- Waypoints—the objects which specify a place. These can be used to designate special areas for developers, or allow creatures to walk a pre-determined path.



Center: a number of different types of objects placed in the world.
Right: the Object Painter, which categorizes Object types with easily identifiable icons.

Graphical Programming

Once content has been placed in an area, the developer is charged with making the environment interactive. One way which the world builders can bring life to an area is by adding dialog. The Aurora Toolset makes giving a character dialog relatively simple. By selecting the Conversation attribute in a Creature's properties (only Creatures can be given dialog), developers are able to edit the dialog that a player can have with that creature. The Conversation Editor makes creating conversations simple, by following a simple one-to-one speaking ratio—first, the player is greeted by the creature, then the player is able to make a response, and the conversation returns to the creature. Each line in the conversation is represented as a dialog node, thus allowing multiple lines to be spoken by the player or creature at any given time. To grant a player choices in the dialog, writers simply need to add a new line as a sibling to any other options the player should have in that conversation. To provide writers with a mechanism to allow a conversation option to return to a previous point or jump ahead to a new place in the dialog, the Aurora Toolset has the option of pasting another line of dialog as a link under a conversation line. Thus, conversations where the player is allowed to ask a series of questions before continuing in the conversation are easy to create.



```
Root
  [OWNER] - So, Mr. <LastName>, I hear there's been some mistake. Apparently, you are not dead.
    That's a relief.
      [OWNER] - For you, maybe. For me, I have to ask how it happened.
        [CONTINUE]
          [OWNER] - Fortunately, I can fix this. However, I was hoping you could do me a favor before I do.
            Are you serious?
              [OWNER] - I am.
                [CONTINUE]
                  [OWNER] - I heard how you dealt with The Refused. Very impressive. We have our set of troubles inside the walls as well.
                    [CONTINUE]
                      [OWNER] - There's a gang known as the Furies. I need you to deal with them.
                        Who are the Furies?
                          [OWNER] - They are a group of terrorists operating inside the city. They're nothing more than anarchists trying to destabilize my Empire.
                            [Return]
                              [OWNER] - There's a gang known as the Furies. I need you to deal with them.
                        So how did this all happen?
                          [OWNER] - No idea. This sort of mix up doesn't usually happen. But, sometimes it does.
                            [Return][END DIALOGUE]
                        Why can't you just send me back now?
                          [OWNER] - Well, it'll take a day to get everything set up, so you might as well do me a favor while you wait.
                            [Return][END DIALOGUE]
                        [CONTINUE]
                          [OWNER] - The Furies have made established themselves in the sewers beneath this city. Bring me their leader, and I will help you in reterning to your world.
                            [END DIALOGUE]
            What do you want me to do?
              [OWNER] - I heard how you dealt with The Refused. Very impressive. We have our set of troubles inside the walls as well.
            No.
              [OWNER] - I was being polite. You will be doing me a favor before I help you.
                [CONTINUE]
                  [OWNER] - I heard how you dealt with The Refused. Very impressive. We have our set of troubles inside the walls as well.
    Ok.
    So, why am I here?
  [OWNER] - Retrieve for me Jabber, the leader of the Furies. His gang has made camp in the sewers below Dis. Bring him back to me alive, and I will help you get home.
  [OWNER] - Excellent work apprehending the fiend. We're almost done setting up for the ceremony. If you want to just get some rest, we'll be ready when you wake up.
  [OWNER] - Are you ready to go back?
```

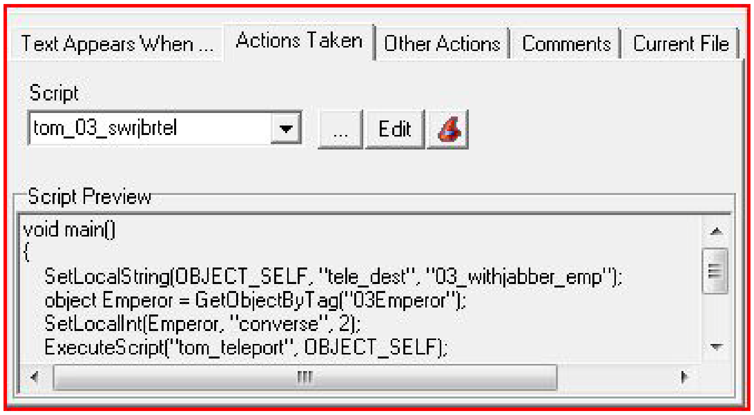
Line Letter Count : 83 Line Word Count : 14 File Word Count : 401 Module Word Count : 3468

So, Mr. <LastName>, I hear there's been some mistake. Apparently, you are not dead.

```
int StartingConditional()
{
  int converse = GetLocalInt(OBJECT_SELF, "converse");
  return converse == 0;
}
```

The Conversation Editor can be used to make simple scenarios or complex dialog scenes

In addition to dialog, the Conversation Editor also provides a means to accomplish a number of common actions in the *Neverwinter Nights* world. Such actions, like playing a sound, updating the Journal of a player, or playing an animation, can all be performed by selecting options in a drop-down box in the Other Actions tab of the conversation editor. Other actions, like giving or taking from the player an item or gold, starting a fight, or starting a sale can also be performed from the conversation editor, using the Actions Taken tab. Each of these actions creates a script in the Actions Taken attribute of a particular line of dialog, which is open to the developer to modify to suit any additional needs. Finally, if the author only wants particular dialog options to be conditionally available, the Aurora Toolset provides a number of pre-made conditions, such as player class, gender, race, whether they have a particular item, or based on a random number, in the Text Appears When… tab. Like the Actions Taken tab, these conditions are created as new scripts, which developers can later modify. With these tools readily available to Aurora Toolset users, writing dialog and performing common in-game tasks can be done easily with some basic instruction.

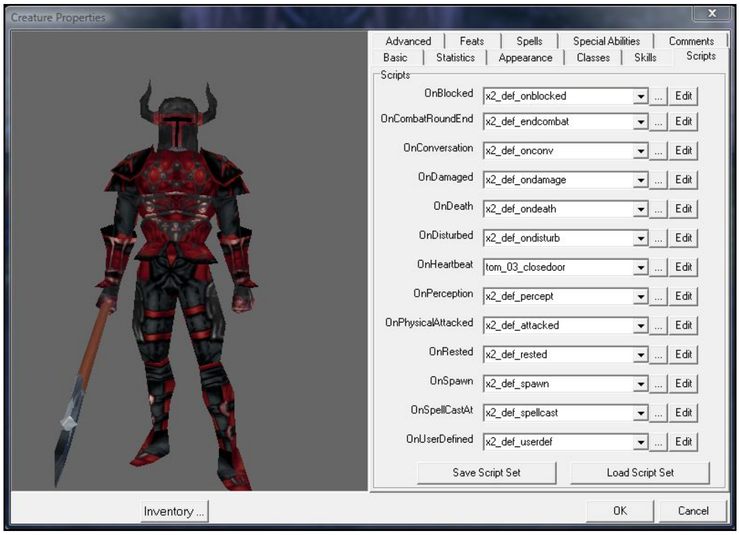


Script: tom_03_swrjbrtel

```
void main()
{
   SetLocalString(OBJECT_SELF, "tele_dest", "03_withjabber_emp");
   object Emperor = GetObjectByTag("03Emperor");
   SetLocalInt(Emperor, "converse", 2);
   ExecuteScript("tom_teleport", OBJECT_SELF);
```

The action panel can be used to perform simple tasks, like giving the player gold, or more complicated ones which effect objects around your world.

Outside of the dialog editor, developers can give special actions to objects and creatures through the use of the Creature Properties panel. This panel grants developers access to many of the internal workings of the creature or object by allowing them to alter the race, faction, combat statistics and capabilities, appearance, and sound sets, among other details. Additionally, each object has a set of preset scripts which determine how it reacts to the player and other objects in the environment. Each of these can be altered, or users of the toolset can change the script reference all together to allow for new and unique behaviors. Scripts can be changed on the object instance level or for every instance of that object by modifying the object blueprint. These settings are not limited to objects, as the module and areas have a set of scripts which can be modified or changed altogether. By modifying area, module, and object scripts, developers can extend the capabilities of the toolset and game engine to create new experiences in the *Neverwinter Nights* world.



Scripts

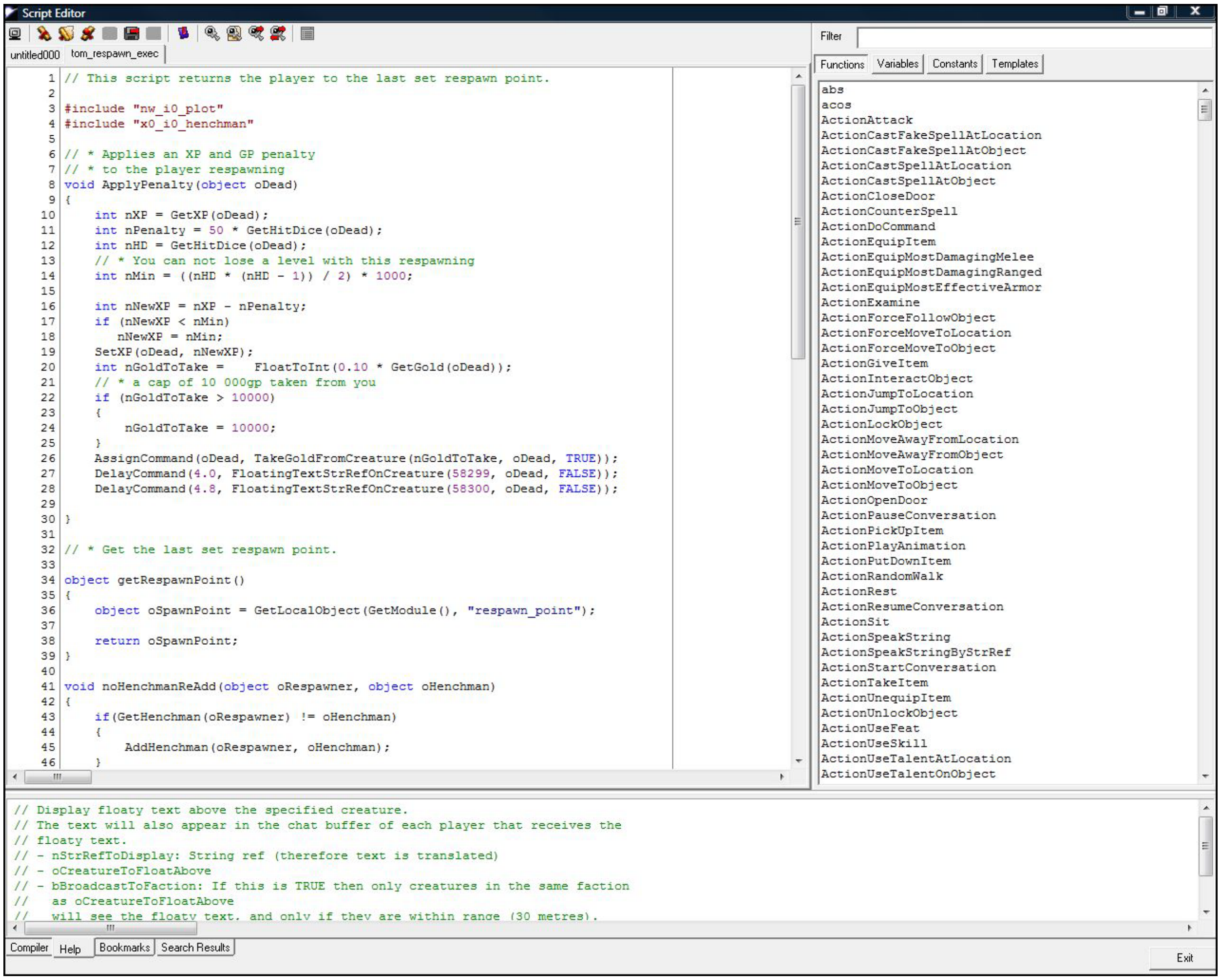
OnBlocked	x2_def_onblocked
OnCombatRoundEnd	x2_def_endcombat
OnConversation	x2_def_onconv
OnDamaged	x2_def_ondamage
OnDeath	x2_def_ondeath
OnDisturbed	x2_def_ondisturb
OnHeartbeat	tom_03_closedoor
OnPerception	x2_def_percept
OnPhysicalAttacked	x2_def_attacked
OnRested	x2_def_rested
OnSpawn	x2_def_spawn
OnSpellCastAt	x2_def_spellcast
OnUserDefined	x2_def_userdef

The Creature Properties panel, with a view of all the scripts applied to this Guard of Dis.

NWN Script

As seen above, Creatures, Areas, and even the Module can be extended using premade or custom scripts. By default, *Neverwinter Nights* is shipped with over 4000 premade scripts, all of which are available to the developers. These scripts range from default actions for NPCs and Monsters in combat situations to system-level details, like how death, re-spawn, and equipping are handled. Many, if not all, of the scripts that come prepackaged with *Neverwinter Nights* were used in developing the main campaign and the expansions that followed. By recognizing this, developers with experience in the *Neverwinter Nights* world can find scripts that were used in-game to achieve interesting or desirable results, and use those same scripts with modifications or in their original forms to mimic scenarios. With the sheer number of scripts available in the Aurora Toolset on install, developers have a ready arsenal to begin developing their new modules.

Despite the incredible number of scripts that come with the Aurora Toolset, developers may choose to create their own scripts to extend the game-play beyond the core features of *Neverwinter Nights*. To do this, the Aurora Toolset provides a powerful Script Editor, which allows developers to view any scripts that exist within the module or the *Neverwinter Nights* campaigns, and create new scripts to add to the module. The editor provides access to a significant number of functions that are commonly used and are specific to the *Neverwinter Nights* game world, such as AssignCommand, which allows developers to tell particular objects to perform an action specified, or ActionRandomWalk, which forces a creature to wander around its placed region aimlessly until another action is assigned to it. Each of these functions are displayed openly on a side panel in the editor, which allows developers to filter the functions by a given string. For example, developers looking for actions that cause a creature to move can type “move” in the filter box, and all of functions with the phrase “move” will appear. By selecting a function, developers can view a description of the function, as well as the parameters and a brief explanation of what each represents. Together, these two sections of the editor make creation of new script assets simple and easy to understand.



```
// This script returns the player to the last set respawn point.

#include "nw_i0_plot"
#include "x0_i0_henchman"

// * Applies an XP and GP penalty
// * to the player respawning
void ApplyPenalty(object oDead)
{
    int nXP = GetXP(oDead);
    int nPenalty = 50 * GetHitDice(oDead);
    int nHD = GetHitDice(oDead);
    // * You can not lose a level with this respawning
    int nMin = ((nHD * (nHD - 1)) / 2) * 1000;

    int nNewXP = nXP - nPenalty;
    if (nNewXP < nMin)
       nNewXP = nMin;
    SetXP(oDead, nNewXP);
    int nGoldToTake =    FloatToInt(0.10 * GetGold(oDead));
    // * a cap of 10 000gp taken from you
    if (nGoldToTake > 10000)
    {
        nGoldToTake = 10000;
    }
    AssignCommand(oDead, TakeGoldFromCreature(nGoldToTake, oDead, TRUE));
    DelayCommand(4.0, FloatingTextStrRefOnCreature(58299, oDead, FALSE));
    DelayCommand(4.8, FloatingTextStrRefOnCreature(58300, oDead, FALSE));

}

// * Get the last set respawn point.

object getRespawnPoint()
{
    object oSpawnPoint = GetLocalObject(GetModule(), "respawn_point");

    return oSpawnPoint;
}

void noHenchmanReAdd(object oRespawner, object oHenchman)
{
    if(GetHenchman(oRespawner) != oHenchman)
    {
        AddHenchman(oRespawner, oHenchman);
    }
```

```
abs
acos
ActionAttack
ActionCastFakeSpellAtLocation
ActionCastFakeSpellAtObject
ActionCastSpellAtLocation
ActionCastSpellAtObject
ActionCloseDoor
ActionCounterSpell
ActionDoCommand
ActionEquipItem
ActionEquipMostDamagingMelee
ActionEquipMostDamagingRanged
ActionEquipMostEffectiveArmor
ActionExamine
ActionForceFollowObject
ActionForceMoveToLocation
ActionForceMoveToObject
ActionGiveItem
ActionInteractObject
ActionJumpToLocation
ActionJumpToObject
ActionLockObject
ActionMoveAwayFromLocation
ActionMoveAwayFromObject
ActionMoveToLocation
ActionMoveToObject
ActionOpenDoor
ActionPauseConversation
ActionPickUpItem
ActionPlayAnimation
ActionPutDownItem
ActionRandomWalk
ActionRest
ActionResumeConversation
ActionSit
ActionSpeakString
ActionSpeakStringByStrRef
ActionStartConversation
ActionTakeItem
ActionUnequipItem
ActionUnlockObject
ActionUseFeat
ActionUseSkill
ActionUseTalentAtLocation
ActionUseTalentOnObject
```

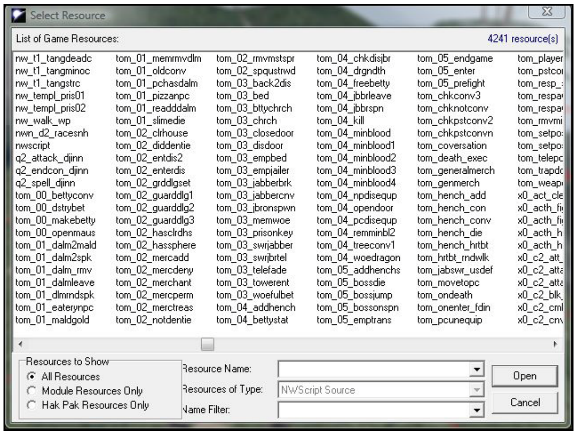
```
// Display floaty text above the specified creature.
// The text will also appear in the chat buffer of each player that receives the
// floaty text.
// - nStrRefToDisplay: String ref (therefore text is translated)
// - oCreatureToFloatAbove
// - bBroadcastToFaction: If this is TRUE then only creatures in the same faction
//   as oCreatureToFloatAbove
//   will see the floaty text, and only if they are within range (30 metres).
```

The Script Editor, with the script being displayed on the left, the Function tab on the right, and a definition of the function and its use on the bottom

Of equal significance is the Constants tab, which, similar to the functions tab, allows developers to see all of the constants defined in the *Neverwinter Nights* realm, and filter them by key terms. Constants are used to define many kinds of values, from colors and object types to animations and special effects. This is particularly useful, as many functions take arguments that are defined as constants for easy identification. For example, the function “SetFogColor” has 3 parameters, two of which are fog type and fog color, while the third is the area to set the fog color for. Developers can pass any number into the first two parameters, but these numbers would have little meaning. However, by looking up FOG_TYPE in the Constants tab, developers would see the different types of fog available. Secondly, by searching for FOG_COLOR, the developer would gain access to a number of pre-defined colors, like BLACK, BLUE, or RED_DARK, which are much more easily understood than the hexadecimal value traditionally used to define the color. With assistance from the Functions and Constants tabs, developers will find the Aurora Toolset’s Script Editor to be a viable tool in preparing a new module for *Neverwinter Nights*.

Shortcomings

Where the Aurora Toolset succeeds in its simplicity, it falls short in a few areas. Of particular note is a lack of consistency in the ways that various kinds of assets are managed. While art assets, like creatures, items, or placeables, are all organized in a tree-like structure based on categories, technical assets, like dialog and scripts, are listed in a flat structure. This issue becomes even more relevant when developers hit the name length cap of 16 characters on technical assets. As such, developers seeking organization in their technical assets are forced to include categorization in the name, but then lose the ability to clearly express the intent of the script. One can see the implications of this issue when viewing the list of 4000 scripts that are included with the toolset, each of which has names no more or less revealing than "nw_cb_lv203." If the toolset instead provided a categorization similar to the packages system found in modern object-oriented languages, the above script instead could have had a name like "nwn.combat.wizard.spells.castspells," which much more clearly identifies the intent of the script. This categorization would also benefit developers in their search for reusable assets, as they would be able to search through the nwn.combat.wizard directory to find all scripts relevant to wizard combat. By adding script categorization, the Aurora Toolset and its included assets would become significantly more capable to the developers using it.



List of Game Resources: 4241 resource(s)

```
nw_t1_tangdeadc    tom_01_memrmvdlm   tom_02_rmvmstspr   tom_04_chkdisjbr    tom_05_endgame       tom_player
nw_t1_tangminoc    tom_01_oldconv     tom_02_spqustrwd   tom_04_drgndth      tom_05_enter         tom_pstco
nw_t1_tangstrc     tom_01_pchasdalm   tom_03_back2dis    tom_04_freebetty    tom_05_prefight      tom_resp_
nw_templ_pris01    tom_01_pizzanpc    tom_03_bed         tom_04_jbbrleave    tom_chkconv3         tom_respa
nw_templ_pris02    tom_01_readddalm   tom_03_bttychrch   tom_04_jbbrspn      tom_chknotconv       tom_respa
nw_walk_wp         tom_01_slimedie    tom_03_chrch       tom_04_kill         tom_chkpstconv2      tom_rmvmi
nwn_d2_racesnh     tom_02_clrhouse    tom_03_closedoor   tom_04_minblood     tom_chkpstconvn      tom_setpo
nwscript           tom_02_diddentie   tom_03_disdoor     tom_04_minblood1    tom_coversation      tom_setpo
q2_attack_djinn    tom_02_entdis2     tom_03_empbed      tom_04_minblood2    tom_death_exec       tom_telepc
q2_endcon_djinn    tom_02_enterdis    tom_03_empjailer   tom_04_minblood3    tom_generalmerch     tom_trapdc
q2_spell_djinn     tom_02_grddlgset   tom_03_jabberbrk   tom_04_minblood4    tom_genmerch         tom_weap
tom_00_bettyconv   tom_02_guarddlg1   tom_03_jabbercnv   tom_04_npdisequp    tom_hench_add        x0_act_cle
tom_00_dstrybet    tom_02_guarddlg2   tom_03_jbronspwn   tom_04_opendoor     tom_hench_con        x0_acth_fi
tom_00_makebetty   tom_02_guarddlg3   tom_03_memwoe      tom_04_pcdisequp    tom_hench_conv       x0_acth_fi
tom_00_openmaus    tom_02_hasclrdhs   tom_03_prisonkey   tom_04_remminbl2    tom_hench_die        x0_acth_h
tom_01_dalm2mald   tom_02_hassphere   tom_03_swrjabber   tom_04_treeconv1    tom_hench_hrtbt      x0_acth_h
tom_01_dalm2spk    tom_02_mercadd     tom_03_swrjbrtel   tom_04_woedragon    tom_hrtbt_rndwlk     x0_c2_att_
tom_01_dalm_rmv    tom_02_mercdeny    tom_03_telefade    tom_05_addhenchs    tom_jabswr_usdef     x0_c2_atta
tom_01_dalmleave   tom_02_merchant    tom_03_towerent    tom_05_bossdie      tom_movetopc         x0_c2_atta
tom_01_dlmrndspk   tom_02_mercperm    tom_03_woefulbet   tom_05_bossjump     tom_ondeath          x0_c2_blk_
tom_01_eaterynpc   tom_02_merctreas   tom_04_addhench    tom_05_bossonspn    tom_onenter_fdin     x0_c2_cml
tom_01_maldgold    tom_02_notdentie   tom_04_bettystat   tom_05_emptrans     tom_pcunequip        x0_c2_cnv
```

Resources to Show
- ● All Resources
- ○ Module Resources Only
- ○ Hak Pak Resources Only

Resource Name:

Resources of Type: NWScript Source

Name Filter:

The script selection window. All 4241 scripts are listed (including the many scripts used to create the Dis *module.*

Disputably, another shortcoming of the tool is the lack of access it provides to the lower level functionality of the *Neverwinter Nights* engine. Scripting is done simply by accessing the high level function API to perform basic actions, like ActionRandomWalk or ActionMoveToObject. Instead, it could have been beneficial to be able to delve deeper into the engine, to see what many of the actions were composed of, and produce new actions that operated differently from the standard actions provided to Aurora Toolset developers. However, this access could be dangerous to inexperienced developers, and certainly reveals what could be considered trade secrets of Bioware. As such, this statement is made merely as a note of intrigue, rather than a complaint against the decisions made by the toolset developers.

What We'd Do Differently

Story

For those who are planning to tell a story using the Aurora Toolset it is important to realize that it is only designed to create *Neverwinter Nights* clones. It is almost impossible to modify the game-play in any meaningful way. The toolset goes so far as to prevent developers from adding spells to the game. Instead, developers are left with the ability to marginally modify existing ones. So, it is important that story-tellers write a *Neverwinter Nights* story. I did not at first, and while I tried to get it to work it never did. The story needs to have a main story line told through a series of connected quests. You can then flush out the world with side quests. Also putting dialog in requires some basic scripting knowledge. If you're good on the tech side you'll be fine. If you have the patience to figure it out you may be fine. If you're like me and you can't stand anything beyond *hello world* give it to your tech guy to do. I wish I had done so earlier.

Art

The Neverwinter Nights toolset comes with a large amount of innate content. Any tileset, creature, or placeable that exists in the standard campaign can be modified for use in a custom module. The toolset also provides a way to get one's own custom art files into the engine, but the act of doing so is a major hassle and time sink. The most compatable 3D modeling program to use is 3D Studio Max (Version 5), which is quite old. The export scripts that are used to convert the .3ds files into valid .mdl files are usable, but can not handle translating certain functions within 3D Studio Max. For example, any object that uses the mirror function or boolean operations will not show up properly when imported into *Neverwinter Nights*. Various material properties such as object transparencies must be done in exactly the right way before they show up properly in the *Neverwinter Nights* engine. This leads to attempt after attempt of failed content creation, and eventually forces the artist to stick with very simple objects that have a minimal chance of failing once imported into the engine.

As the idea of making new content from scratch becomes less desirable, the idea of using preexisting objects becomes much more appealing. Learning to navigate all the content in the engine and in *Neverwinter Nights* Explorer is a must. Had I become familiar with all of the assets I had available to me sooner, I would have gone about making them differently. Some of the placeables I made would have been easier (and would have turned out better) if I had based them off of existing *Neverwinter Nights* content, instead of fighting with 3DS Max and its export scripts.

Even after learning all the intricacies of the many tools and programs required to make this module, a large amount of time must be invested to get anything significant done. If the production of the module falls behind, or an earlier part of the game is revised, then the entire team must work harder to make up for it. However, it is the artist who has to spend the most hours to catch up. My biggest regret with this module is simply not having enough time to incorporate all of the features I had in mind. Delays in the design process caused delays in the art production for the later areas of the module, which led to features eventually being dropped. Perhaps our team's goals were a bit too lofty (instead of making the recommended 1 hour module we made something closer to 3 hours).

If I were to do this project again, I would suggest making a much shorter module, with fewer but more detailed areas, and more interaction with npc's and the environment. I would also be more active in keeping the team on schedule, because I have learned that when one person falls behind, I was the one that had to work the hardest (or atleast the longest) to make it up.

Technical Development

Although the project group has successfully produced a working module for *Neverwinter Nights*, a few issues were encountered along the way. Perhaps the most severe was the issue of getting dialog into the game. Upon beginning the project, the team was informed that dialog would most effectively be written in the game engine, using the conversation editor tool, as opposed to other standard word processing tools. However, due to the nature of the dialog and the events occurring within it, it slowly became clear that the Conversation Editor would not be the most viable tool for initially creating the dialog sequences. Although the tool provides many simple wizards for creating basic actions, many of the actions involved in the dialog sequences were too complicated to be represented properly. Additionally, the dialog often called for multiple responses from NPCs, which increased the technical component of preparing the dialog. As such, it has been determined that the best course of action would have been to have the dialog prepared separately from the Aurora Toolset, then added and implemented later by a member of the team with more technical experience. Had this been done from the onset of the project, scenario scripting could have begun months before it did, resulting in a cleaner, more developed module.

Additionally, throughout the development of many of the technical components of the module, a few problems manifested, many of which were due to the use of modified prepackaged scripts. After investigation, it became clear that many of the 4000 scripts included with the Aurora Toolset were very task specific, and included many nuances that would only manifest themselves in a default *Neverwinter Nights* campaign. However, given the number of altered systems in the *Dis* module, such as the checkpoint system, many systems that performed adequately in the default *Neverwinter Nights* campaigns no longer functioned in the context of the new module. The greatest culprit was the henchmen system, which was used to provide the player with additional companions throughout the module, and its interactions with the checkpoint system. Many hours were spent attempting to rectify the compatibility issues found in these systems. Although a solution was found, we have concluded that the standard henchmen scripts were not suited to our needs, and custom scripts should have been developed. By creating custom scripts to produce the appropriate actions for henchmen, more dynamic henchmen could have manifested (for example, Dalm, the henchman the player encounters in the first caves dungeon, is able to make comments based on the situations the player is in), and the henchmen system may have more readily integrated with the checkpoint system.

Conclusion

The Aurora Toolset is great at doing what it is supposed to do, but little else. Every member of the team experienced issues in some form regarding this. As far as problems are concerned this is not a significant one. The art pipeline can turn into a huge problem if you let it. For the most part we were able to succeed in making the mod we wanted to make. Probably the biggest issues were faced by our lead designer and writer, but most of that was plain stubbornness. It is not worth fighting the engine. Once we sat down and figured out the engine, we were able to go about designing a game that actually worked with it. Once we did that, we were able to make a fairly decent game and were able to tell a sophisticated story, build interesting environments and creatures, and there’s even some clever code thrown in there. The toolset will determine what type of game you’ll make, but you can still use it to make a good game.

Appendix A: Design Documents

Vision Statement

Dis

Dis is a visionary, forward thinking Role Playing Game. The game play elements of *Dis* would be recognizable by any fan of the genre held together by a story that might be expected from a Hollywood Movie.

High Concept

The main character isn't a superhero, but soon discovers a power he wasn't aware he possessed.

When the game begins he is before the graves of his recently deceased wife and daughter, both of whom were lost in an accident caused by the main character. While the main character deals with the trauma he discovers another world that may offer him hope to regain his lost family.

The Game

***Dis* utilizes the same game engine as Bioware's acclaimed and highly successful *Neverwinter Nights* and its subsequent expansions.**

The D&D rules have withstood the tests of time. As a game mechanic they have proven themselves to be both fun and intuitive, elegantly hiding complex calculations behind a simple user interface that even the most inexperienced user can quickly grasp.

Game Play

Much of the game is an expression of the main character's inner demons. These are jailers from which not even death grants release.

A new world waits hidden in a labyrinth beneath the main character's home. This world is a sprawling fantasy realm, featuring villains, allies, betrayal and hope.

As stated above the game uses the D&D rule book. This allows for simple point and click game mechanics. In addition, we will create interesting tactical choices. Enemies will have weaknesses and strengths. Ignore these and the player will find even the most basic encounters dangerous; exploit them and the player can blow through most fights.

Story

Audience

The game play is designed to appeal to the fans of D&D role playing games while still making provisions for inexperienced players.

The mature, complex story will draw in individuals who are generally not drawn to the genre as well as those who only casually play games. In particular, the target are fans of the video game franchise *Silent Hill*, as well as movie goers who enjoyed *Rosemary's Baby* or *Jacob's Ladder*, or even the symbolist poetry of Charles Baudelaire.

By the end of the game, the player will have experienced one day in the life of a tormented man doomed to repeat it once again tomorrow.

In the end this is a D&D game, and the story is crafted around that experience.

The story is set in a fantasy realm. It is a violent world filled with violence and betrayal. At the same time, you will find loyal allies and most importantly, hope that the main character can restore his family.

Basic Outline

Level one – House
Start: Grave Yard
Enemies: Weak Spirits
Quests: Ready House for Hurricane
Enter Caves
Memories: Church

Level two– Caves Level One
Start: Cave entrance
Enemies: The refused
Quests: Locate Dis
Memories: Eatery
Lombardi's

Level three– Dis
Start: Dis
Enemies: Furries
Fire Spirit
Quests: Rescue Daughter
Memories: Home (daughter)

Level four– Caves Level Two
Start: Dis
Enemies: Harpies
Quests: Save Betty
Memories: Water (Player must Drown)

Level Five– Home
Start: Grave Yard
Enemies: Storm Spirits
Quests: Return to Caves
Rescue Family
Memories: Lost (the Faceless)

Detailed Outline

Level One

House on a lake with a small grave yard in the back. It is raining with a large storm approaching. The house is norm except for the presence of some weak spirits.

Quests:

Barricade Home

The player must barricade his home for the coming storm.

Memories:

Church

Player finds a photo from his wedding. When he leaves the room he is in the church remembering his wedding. He exits the church into his basement. The stairs have collapsed and a deep hole has formed in the ground.

Level Two

The player lands in a series of caves. An old man greets him. Here the player chooses his "spirit self". This will be the character the player plays as for most of the game. The caves exit onto *The Scalded Plains*. Bandits known as *the refused* roam the plains as well as several carnivorous beasts. A little ways in the player comes to a mostly abandoned village. Here he can recruit a henchman. Eventually the plains end at the walls of Dis. Outside refugees await entrance to *Dis*, preyed upon by *the refused*.

Quests:

Gain Entrance to Dis

The old man knows the player doesn't belong in this realm however there is nothing he can do to help; only the emperor of *Dis* can help. To reach Dis the player must cross the *Scalded Plains*. When he reaches Dis he learns the gates are closed because the reused have been hiding within the refugees. To enter the player must kill the leader of the Refused and eliminate any refused hiding among the refugees.

Henchman

The quests vary for each henchman but all require entrance to Dis.

Memories:

Pizzeria and Eatery

In the caves the player may discover the pizzeria and eatery. Inside the memory plays out.

Level Three

Dis is dirty overflowing with people. A large maze of a city, with winding streets and catacombs arranged around a pair of twin towers. The first, the Tower of Judgement, is the seat of power home to the Emperor and his court. The second is mysteriously known as the Reflecting Pool. Roaming the streets are mostly peasants, merchants and swarms of refugees. However rumors hold that the Refused have gained a foothold in the city and an even more dangerous rebel group known as the *Furries* hide in the shadows.

Quests:

The Reflecting Pool

Upon entering the city the Player here's the voice of his daughter. She beckons him to the Reflecting Pool. There the Player is confronted by the memory of his daughter. Inside the tower is arranged like the players home. As he enters each room they come alive with a memory of his daughter. Discovering them to be imitations, the distraught player burns the images. Atop the tower the player finally finds his daughter who is held captive by the Reaper. Afterward the Player must escape through the burning tower with his daughter. Outside she informs the player Betty also exists in this world.

Quests:

The Refused

Words of your deeds have entered the walls of Dis. The refused will attack the player as he wanders the streets. Clear they won't stop until he is dead the player decides their leadership must be destroyed here as well.

The Emperor

According to the old man the player doesn't belong but only the ruler of Dis knows for sure. The Emperor knows of the player but will not grant an audience until the player performs a task for him. Apparently there is a rebel group know as the Furies plotting against Dis. Defeat them and the Player will be granted an audience.

According to the Emperor the player's soul has returned to the Spirit realm of its own will. While the body isn't dead his spirit won't return until it has completed its task in this realm. Knowing the task will be dangerous the emperor offers to protect the player's daughter within the walls of the Tower.

Level Four

That night Betty approaches the player calling for his help. She is trapped somewhere in the *Woeful Woods* which stretch beyond the walls of Dis. The woods are home to a flock of harpies and the Emperors personal prison. The souls subject to punishment are transformed into immortal trees fed. Each day the harpies strip them and each night they grow back. Betty is here trapped as one of the trees.

Quests:

Rescue Betty

Betty awaits the player in the woods; however, after some research it becomes clear that just reaching her will not be enough. Apparently only the blood of the Minotaur can reverse her curse. The creature lives beneath Dis in its sprawling Catacombs. The player needs a vile of blood; he can either gain it by slaying the Minotaur or stealing it from the Imperial Alchemy. With the blood he must enter the Woeful Woods and free Betty.

Escape Geryon

In addition to the harpies roaming the woods the Emperor has also placed the dragon Geryon. The dragon is massive and possesses the intelligence of a man. He is a competent jailer who attacks the moment Betty is released. The player may attempt to fight Geryon but finding it hopeless he must run with Betty. As the gates of Dis are barred to them there only chance of escape is into Acheron the ocean which surrounds the woeful woods.

Memory:

Water

The player and Betty escape into the water trapped with Geryon above they must fight off water spirits as the player slowly drowns.

Level Five

The Player awakes in the graveyard behind his home. The rain is pouring down. The weak spirits are now Storm Spirits. The house is collapsing, only a path to the basement is clear. In the basement the pit remains but now it is bigger and deeper.

The player lands in a large plain of ice. The faceless lost roam the fields. In the center is a sister tower of the twin towers of Dis: *Lucifer*.

Quests:

Rescue Your Family

Your henchman meets you when you arrive. He rushed back to Dis in an attempt to rescue your daughter but it was hopeless as she was already in the Emperor's grasp. Shortly Geryon returned with Betty. Your Henchman had to go into hiding, trying to locate them and waiting for your return. Now your only choice is to climb Lucifer and save them.

Appendix B: Dialog

Home

Main Quest

Player starts in the backyard of his house. Betty's ghost stands by the gravestones. The mausoleum is locked.

BETTY
What's the matter?

PLAYER
1. Nothing.
2. I miss you.
3. ...

BETTY
1,3. It's aright.
2. You're sweet.

PLAYER
1. Continue

BETTY
There's nothing you can do. Just get some sleep, you'll be better in the morning.

PLAYER
1. Exit

The mausoleum is now unlocked. If you absolutely must put a damn portal in that closes behind the player trapping him. It doesn't really matter how he gets to Dis just that he does.

The Caves

Main Quest

The Player emerges in the caves, immediately he is jumped by a group of bandits. After the player defeats them an old man reveals himself.

OLD MAN
Filthy bandits, anyway good job, I would've dealt with them myself but I'm more of a lover than a fighter. Anyway, name?

PLAYER
1. What?
2. Where am I?
3. Who are you?

OLD MAN

1. Your name, what’s your name.
2,3. Your name first then we can talk.

PLAYER
1. <FullName>

OLD MAN
That’s not right. What are you doing here?

PLAYER
1. I don’t even know where here is.

OLD MAN
Alright there’s been some mistake. You’ll need to see the Emperor; he’s the only one who can sort this out.

PLAYER
1. What’s going on?

OLD MAN
You’re dead, everyone here is dead.

PLAYER
1. What!?

OLD MAN
Great, you’re clueless. Alright, ask your questions.

PLAYER
1. Who are you?
2. Where am I?
3. The Emperor?
4. What the hell do you mean dead!?
5. Never mind.

OLD MAN
1. I’m nobody. I just take care of the clerical work, and make sure the dead get where they belong.
2. These are the caves, one of the entrances to the Empire of Dis, the land of the dead. Only thing is you’re not supposed to be here.
3. The ruler of Dis and governor of souls. He’ll know what’s going on.
4. D-E-A-D like a doorknob. I don’t have a clue how you died, just know that everyone in DIS is dead.
5. Alright, to get to the emperor you need to make your way through the caves, and across the Scalded Plains. You’ll find him in his tower located in the center of Dis.

<u>Spelunkers</u>
The player will start quest when he finds either Dalm in the caves.

.
DALM
Hey dear, could you give me an itsy bit of help.

PLAYER
1. Itsy bit?

DALM
Yeah, I was um... Spelunking, with my friend Mald, but we got in a tiny fight and I ran off. Unfortunately now I'm lost. So, deary, give me a hand getting back to Mald?

PLAYER
1. Yes.
2. No.

DALM
1. Great.
2. *Dalm starts crying*

player returns after saying no.

DALM
sniffling Did you reconsider?

PLAYER
1. Yes.
2. No

DALM
1. Oh you're such a dear.
2. *Starts crying again.*

player reunites Dalm and Mald

MALD
DALM!!

DALM
MALD!!

They embrace

PLAYER
1. Hey do I get some sort of reward.
2. I'll just leave you two alone.

DALM
1. Sure here you go. *(player gets 50 gold)*

2. exit

Memories

Pizzeria

BETTY
You know, you eat here too much.

PLAYER
1. Wait this is where we first met.

BETTY
You should try some new places, there’s a new Eatery on the Boardwalk I want to try.

PLAYER
1. Is this just a memory?

BETTY
Pick me up around eight.

PLAYER
1. Alright

Eatery

BETTY
So I guess that was a bad choice.

PLAYER
1. I’ll just have to pick where we eat next time.

BETTY
Who says they’ll be a next time.

PLAYER
1. Who are you kidding you want me.

Scalded Plains

Main Quest

The player meet’s a guard standing outside of Dis.

GAURD
There’s no entrance to Dis at this time.

PLAYER

1. I need to enter.
2. Why?
3. Great
GAURD
At this time we aren’t allowing entrance to Dis.

PLAYER
1. You already said that.

GAURD
No I didn’t, it was a little different the first time.

PLAYER
1. Sure
2. What if I need to get in.

GAURD
We honestly can’t let anyone in, there’s a group of bandits, The Refused they call themselves. They keep trying to sneak members in with the new souls so until they’re dealt with no entrance.

PLAYER
1. So if I deal with them I can get in.

GAURD
They’re reported to operate out of a cave to the east.

PLAYER
1. exit

talk to him again before completing the quest

GAURD
No luck yet? You might try to get some help. I hear the inn on the hill might have something.

afterwards

GAURD
So you did it, impressive. Well you can enter now.

<u>Hired Hand</u>

The inn is filled with different mercenaries. Each with a different role.

MERCENARY
I’m <name> a pretty powerful <job> if I don’t say so myself. You here for the bounty?

PLAYER
1. Absolutely!

2. What bounty?

MERCENARY
1. There is no bounty, so what are really you doing here?
2. Good answer
PLAYER
1. Continue

MERCENARY
You know you look pretty strong. I can't take all of these guys alone, you want to join up.

PLAYER
1. Sure
2. Team up for what?

MERCENARY
1. Great do you have any idea what I'm talking about?

    PLAYER
    1. None whatsoever.

1,2 That's what I thought, The Refused jumped an Imperial convoy. It's supposedly hidden somewhere on the Scalded Plains, but none of us can find it.

PLAYER
1. Continue

MERCENARY
Of course they probably took all the loot to their hideout.   I know where that is, but I can't get in alone.

PLAYER
1. So where is it?

MERCENARY
Their hideout is in the eastern fields of the Scalded Plains. Obviously head east from here. On the border of the two halves of the Scalded Plains is an old fort.   It's overrun now so you'll need to fight through some refused. Once you do head south til you reach the sea. Follow the shore east and you'll find their cave.

After finding the treasure

MERCENARY
So we did pretty good together, want to make this gig permanent.

PLAYER
1. Sure
2. No thanks
3. Gig, are we in a band.

MERCENARY
1. Great.
2. That too bad.
3. Piss off.

If you talk to him again after saying no.

MERCENARY
Did you reconsider?

PLAYER
1. Yes
2. No

MERCENARY
1. Great.
2. Too bad.

Dis

<u>Main Quest</u>

on Entrance to Dis the player is approached by a Gaurd.

GAURD
Are you <fullname>?

PLAYER
1. Yes.
2. Who’s asking?

GAURD
The Emperor has been waiting for you. Follow me.

PLAYER
1. exit

<u>Quest</u>

Emperor’s Tower

<u>Main Quest</u>

Emperor’s Tower throne room.

EMPEROR
So Mr. <lastname> I hear there’s been some mistake, apparently you are not dead.

PLAYER
1. That’s a relief.
2. Ok
3. So why am I here?

EMPEROR
1. For you maybe, for me, I have to ask how it happened.
2. You seem rather indifferent.
3. Good Question.

PLAYER
1. Continue.

EMPEROR
Fortunately I can fix this. However I was hoping you could do me a favor before I do.

PLAYER
1. Are you serious?
2. What do you want me to do?
3. No.

EMPEROR
1. I am.
3. I was being polite, you will be doing me a favor before I help you.

PLAYER
1. Continue

EMPEROR
I heard how you dealt with the Refused, very impressive. We have our set of troubles inside the walls as well.

PLAYER
1. Continue

EMPEROR
There’s a gang known as the Furies, I need you to deal with them.

PLAYER
1. Who are the Furies?
2. So how did this all happen?
3. Why can’t you just send me back now?
4. Never mind.

EMPEROR
1. Their a group of terrorists operating inside the city. They’re nothing more than anarchists trying to destabilize my Empire.
2. No idea, some sort of mix up usually doesn’t happen, but sometimes it does.

3. Well it'll take a day to get everything set up, so you might as well do me a favor while you wait.
4. exit

after you eliminate the Furies.

EMPEROR
Excellent work, we're almost done setting up for the ceremony, if you want to just get some rest we'll be ready when you wake up.

after memory

EMPEROR
Are you ready to go back?

PLAYER
1. What have you done to Betty?

EMPEROR
I see. That's how you were able to enter my realm. Betty is a traitor, she has been dealt with as our laws require.

PLAYER
1. Continue

EMPEROR
I didn't think the Furies had the power to call you here. In any case we now have Jabber, we'll find out soon enough

PLAYER
1. Continue

EMPEROR
I don't know what your role in all of this is. If you leave now everything will be forgotten. Understand.

PLAYER
1. I'm not leaving

EMPEROR
I see, guards. *The guards come and arrest the player.*

The Player is locked up in a cell beside Jabber.

JABBER
I figured it wouldn't be long before you showed up.

PLAYER

1. Who are you?
2. How do we get out of here?
3. Can I ask you something else?
4. I just want to get out of here.

JABBER
1. I was a priest, you know before I died, obviously this was not what I expected to find in death. That's why I fight, I'm not an anarchist like they claim I am, I'm just against his order.
2. Don't worry I've been captured plenty of times. I'll take care of it.
3. Sure.
PLAYER
1. Do you know how I can escape from Dis?
2. I had a dream...[you describe your dream]
3. Never mind.

JABBER
1. I'm sorry I don't know. I'm not even sure it's possible, the Emperor was probably just using you to get to me.
2. She was calling out to you from the Woeful Woods.
4. Alright just a minute

Jabber opens his cell door and then comes around and opens the player's cell.

JABBER
Betty is in the Woeful Woods. You'll find the woods beyond Dis. However, before you get there you'll want to acquire a vial of Minotaur blood. There may be some stored here in the tower, or you can find the Minotaur hidden deep within the catacombs below the city.

PLAYER
1. Exit.

<u>Memory</u>
While the player rests before ceremony.

Inside a church.

PRIEST
Do you take this man to be your lawfully wedded husband?

BETTY
I do.

PRIEST
And do you take this woman to be your lawfully wedded wife?

PLAYER
1. I do

PRIEST
You may kiss the bride.

The procession heads outside.
Outside Betty and Player are in the Woeful Woods when Betty is attacked by Harpies.
The player awakes

BETTY
Help Me.

PLAYER
1. Exit

Sewers
Player fights through Furies in sewers. He meets and attacks Jabber, the Furies leader, before the player can kill him Jabber surrenders.

JABBER
Alright, that's enough I surrender.

PLAYER
1. Fine, you'll return to the Emperor with me.
2. I don't think so. You'll die here.

JABBER
2. You're new here, you'll see I'm not your enemy. Betty knew that.

Player
1. How do you know Betty?
2. Shut up.

JABBER
1. She worked for me.
    PLAYER
    1. Where is she?
    JABBER
    1. Ask the Emperor.
2. Alright but you should probably ask the Emperor about her.

Woeful Woods

<u>Talking Trees</u>

The player will find Betty (who has been transformed into a tree) at the end of a large forest.

BETTY
Help me.

PLAYER
1. [Use Minotaur Blood]
2. Exit.

BETTY
Thank you.

PLAYER
1. Do you remember who I am?
2. How did this happen?
3. I missed you so much.
4. What is this place?
5. Let’s just get out of here.

BETTY
1. Of course I do silly.
2. ...
3. I know, and I’m sorry, I’m so sorry.
4. It’s the Woeful Woods, it’s one of the prisons in Dis.

> PLAYER
> 1. Prison?
> 2. Never mind.
>
> BETTY
> 1. I was caught spying on the emperor.

5. Alright.

GERYON
Roars

BETTY
We need to hurry.

PLAYER
1. Why.

BETTY
!. Too late.

The Player must engage in a losing fight with Geryon.

<u>Quest</u>

At the entrance of the Woeful Woods a tree begins speaking to the player.

TREE
Is that Minotaur Blood?

PLAYER
1. Yes.

TREE
Please, sprinkle it on my roots.

PLAYER
1. Are you a talking tree?
2. I need this for someone else.
2. [Sprinkle the blood on the trees roots]
3. I think I'll pass

TREE
1. Yes, well sort of, it's a punishment. All the trees in the forest are trapped souls being punished by the Emperor. But the Minotaur Blood frees us, so please sprinkle it on my roots.
  PLAYER
  2. I need this for someone else.
  2. [Sprinkle the blood on the trees roots]
  3. I think I'll pass
2(a) I understand, please if you ever find some more bring it to me.
2(b). [The tree transforms back into a man.]
  PLAYER
  1. Continue.

  TREE
  Thank you so much. I don't have a lot but here take this.
3. What!? Please, please help me.

If the player returns after saying no.

TREE
What do you want?

PLAYER

1. *the appropriate choice 2. from before*
2. Bye

Ice Tower
.
<u>Main Quest</u>

Jabber, Betty, the mercenary, and the player are all on the floor of the ice tower. The Emperor is preparing for some ceremony.

JABBER
Wake up.

PLAYER
1. Where’s Betty?
2. Where are we?
3. Are we still alive?
4. What are you doing in here?
5. What’s next?

JABBER
1. She’s next to you. Don’t worry we’re all fine for now.
2. The Ice Tower, it’s the worst prison in the Empire.
3. Technically? But yeah we’re fine.
4. After your little stunt the Emperor launched a massive hunt for everyone who fought with you. Bastards got me and the mercenary who helped you out in the Scalded Plains.
5. We save the girl and kill the Emperor. That seem about right?

PLAYER
1. Sounds good.

EMPEROR
I was hoping you wouldn’t be awake for this part. I suppose now we’ll have to fight.

He transforms into a monster.
After the fight Betty runs up to the player

JABBER
Well looks like you got the girl.

PLAYER
1. How do we get out of here?

JABBER
Well it’s a long walk back to Dis. But I’m guessing you mean back to your old life. I’m sorry I don’t know.

PLAYER
1. So I’m trapped here.

JABBER
You could try tapping your heels together.

PLAYER
1. Continue.

JABBER
Then again the Emperor’s gone, and Betty’s here, Dis might not be such a bad place to live.

Fin.

Random Dialog

Dis

CITIZEN
Rebels inside the city and bandits outside. I hate to think it but people are starting to doubt the Emperor's power.

CITIZEN
My neighbor up and disappeared the other day. I don't know if he was arrested our kidnapped, you can't trust the rebels or the Emperor.

CITIZEN
A friend of my sister met the leader of the Furies, apparently he's very handsome but never stops talking.

CITIZEN
I fell off of my skateboard in the park. I thought they were just LARPers!

GAURD
The Emperor wants to see you. I don't.

GAURD (after you're assigned to kill the Furies)
Humph...you don't look so tough. I don't know why he won't just let us tear up the town, killing those bastards is our job.

PATROL
What do you want?

PLAYER
1. Tell me about Dis?
2. Where's the Emperor's Tower?
3. What shops are in town?
4. Do you have any clues as to where I can find the Furies?

1. In Dis, we've mastered the art of imprisonment. Minor punishments are served within the prison inside the Emperor's Tower. Crimes committed in your previous life are punished by the harpies in the Woeful Woods. Treason is the highest crime and punishment for it is dealt out in the Ice Tower. Look we don't handle crime lightly so just stay out of trouble.
2. Boy are you blind it's the big thing in the middle of the city.
3. Ginna Borinson runs the only real shop you need. You'll find her at her stall in front of the Emperor's Tower.
4. Oh you're then one the Emperor sent to kill those bastards. We're not sure where their base is, but usually when we chase them off they run north towards the waterfront. I'd look around there for anything that doesn't look right.

GINNA BORINSON
Hey ya. You look like you have need for my wares. Step up and see what I got. I'm sure you'll be pleased.
<u>INN</u>

INNKEEPER
You another refugee? As you can see we're running out of room up here.

PLAYER
1. I'm not a refugee.

INNKEEPER
Then you're one of those damn bounty hunters.

PLAYER
1. I'm looking for a room.
2. Why are all these people here?
3. I'm just trying to get into Dis.

INNKEEPER
1. Really, you've got money for a room?
2. You blind the bandits ran them out of their homes, and of course the Emperor's too busy with counting his money to care.
3. Good luck we're all pretty much locked out of the city. Until these bandits are dealt with we're all going to be treated like criminals.

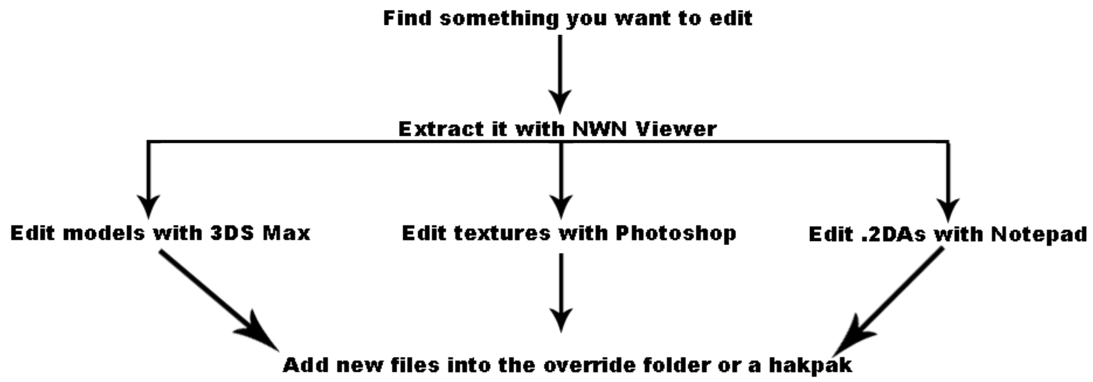
PLAYER
1. I think so. How much for a night?
2. Can I help?
3. I've got to try do you have any supplies that might help?

INNKEEPER
1. 10 gold *or whatever is fair*
    PLAYER
    1. sure
    2. I'll pass
2. Well if you clear the bandits out of the homes I'm sure we could work out some type of reward.
3. Not much but take a look.

Appendix C: Art Walkthroughs

The Art Pipeline: How to Modify/Create Content

Once you are familiar with the process, creating custom content in *Neverwinter Nights* is relatively easy. The basic idea is that you find something in the game that you want to modify, extract it with *Neverwinter Nights* Explorer, modify it, and re-import it into the game.


Find something you want to edit
Extract it with NWN Viewer
Edit models with 3DS Max
Edit textures with Photoshop
Edit .2DAs with Notepad
Add new files into the override folder or a hakpak

The Override Folder Vs Hakpaks

There are two ways to get custom content into a *Neverwinter Nights* Module. The first way is to simply put all of your models, textures, and whatever else into the override folder. The second way is to load all of your files into a hakpak, and apply it to your specific module.

When a *Neverwinter Nights* module is loading, it goes through the process of loading all the default *Neverwinter Nights* files. It then checks to see if there are any files in the override folder, and proceeds to load those. If you have a file with exactly the same name as one of the default *Neverwinter Nights* files in the override directory, the game will load that file instead. This provides an easy way to get custom content into the game, because you can rip content (model, texture, or any other filetype), modify it in some way, and drop it directly into the Override folder. Providing you do not change the file's name, *Neverwinter Nights* will now load your custom content instead of the original.

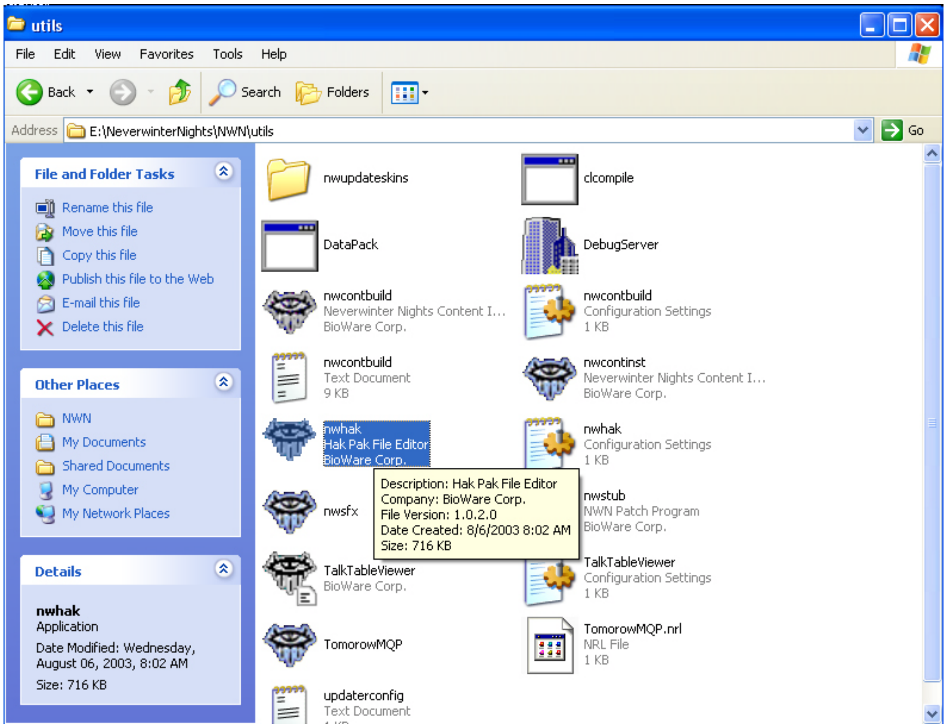
The problem with using this method is that your custom content will replace the original every time, whether it is in your module, another person's module, or the standard *Neverwinter Nights* campaign. If you have only changed a model or texture, this is generally not a big deal, but if you have significantly changed a tileset or .2DA file, you can break the other person's module. These reasons are why it is frowned upon for modules to be released with files in the override directory.

The second way you can get custom content into the game is by using Bioware's Hakpak Editor to create a hakpak. Hakpaks are *Neverwinter Nights*' way of compressing and packaging all the custom content you create. This way allows you to only import your custom content if the specific module calls for it, which avoids the issue simply placing files in the override folder causes.

In general, it is acceptable to use the override folder as you are building a module, but be sure to make everything into a hakpak when the module is finished.

Making Everything into a Hakpak

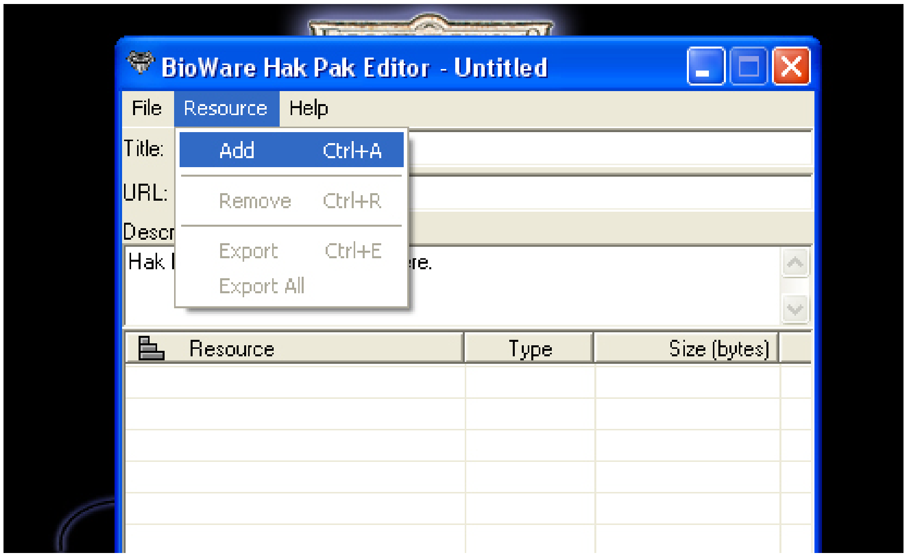
It is very easy to make a hakpak using Bioware's Hakpak Editor. This program comes standard with the *Neverwinter Nights* Toolset, and can be found under the utils directory.



Description: Hak Pak File Editor
Company: BioWare Corp.
File Version: 1.0.2.0
Date Created: 8/6/2003 8:02 AM
Size: 716 KB

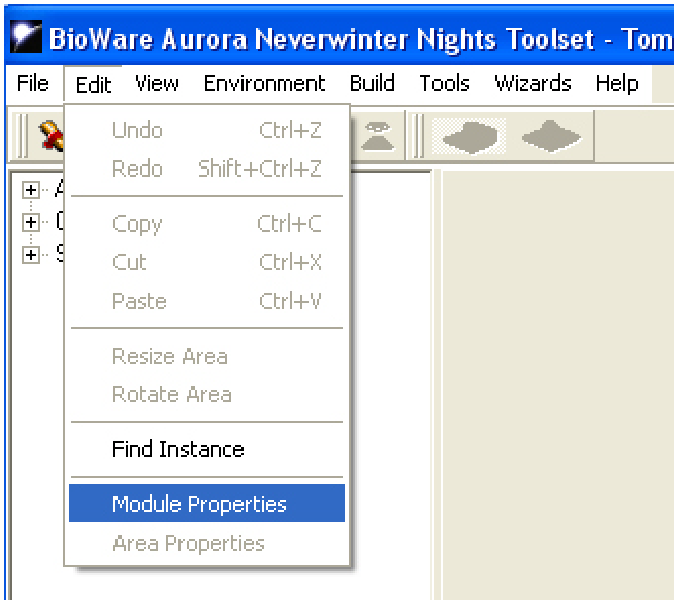
nwhak
Application
Date Modified: Wednesday, August 06, 2003, 8:02 AM
Size: 716 KB

Go to Resource->Add, and then select all of your custom files.

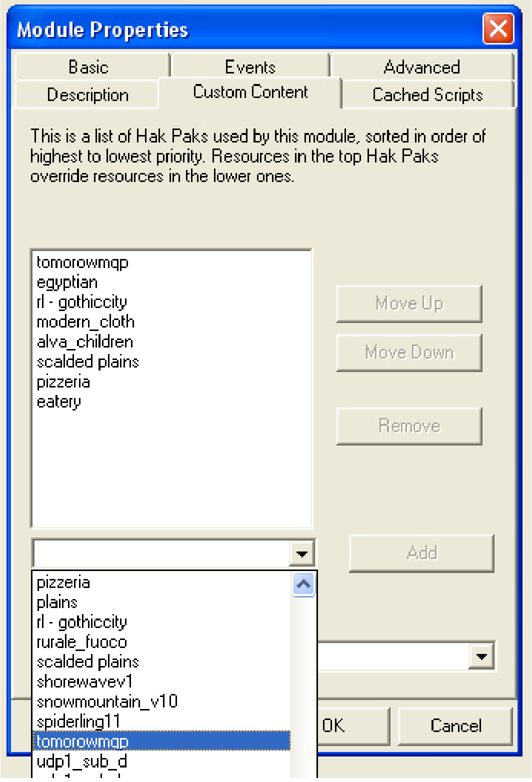


Then click file save and you are done. Be sure to save it in the hak directory.

Now all that's left to do is add your hakpak to your module. Open your module in the Aurora Toolset.



Click Edit -> Module Properties…



This is a list of Hak Paks used by this module, sorted in order of highest to lowest priority. Resources in the top Hak Paks override resources in the lower ones.

tomorowmqp
egyptian
rl - gothiccity
modern_cloth
alva_children
scalded plains
pizzeria
eatery

pizzeria
plains
rl - gothiccity
rurale_fuoco
scalded plains
shorewavev1
snowmountain_v10
spiderling11
tomorowmqp
udp1_sub_d

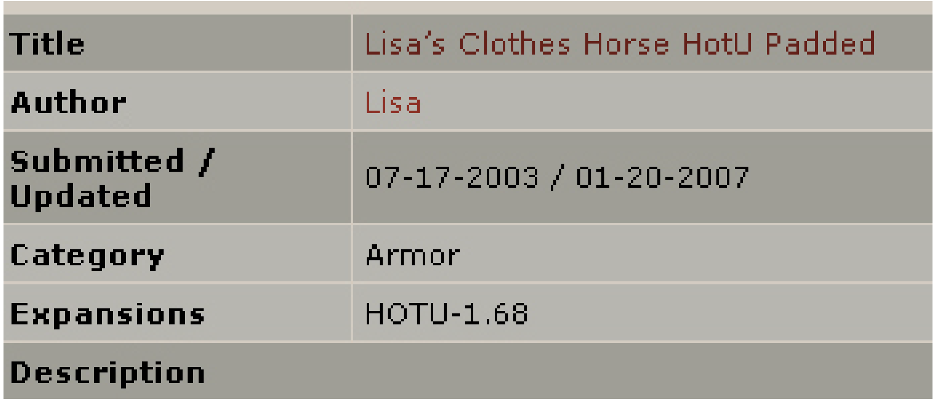
Then go to the Custom Content tab and select your hakpak from the drop down menu.

You can add multiple hakpaks to a module. The order of which they appear on the list only matters if you have multiple hakpaks modifying the same file, in which case the topmost hakpak will overwrite the bottommost hakpak.

Downloading and Importing Custom Content

There is a lot of custom content readily available online, on websites such as IGN's *Neverwinter Nights* Vault. It is easy to download and incorporate this content once you understand how hakpaks work.

For the *Dis* module, we wanted the memory scenes the player experiences to resemble modern times. However, *Neverwinter Nights* did not have any clothing or armor that seemed appropriate for our module. After searching *Neverwinter Nights* Vault, an appropriate custom clothing package was discovered.



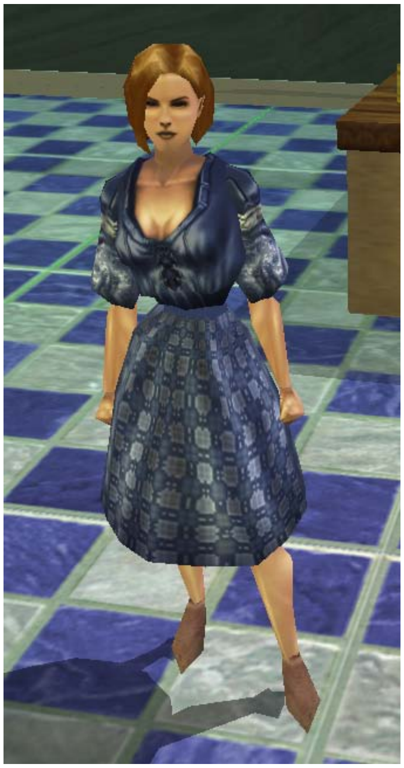
Title	Lisa's Clothes Horse HotU Padded
Author	Lisa
Submitted / Updated	07-17-2003 / 01-20-2007
Category	Armor
Expansions	HOTU-1.68
Description	

Most custom content available online is already in hakpak form, so simply download it, add it to the hak directory, and add it to your module.

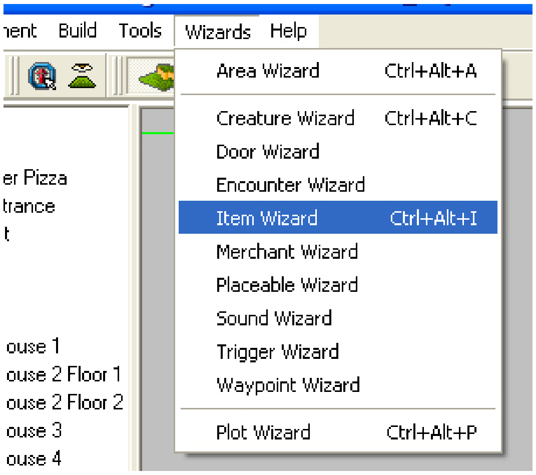
Making Custom Armor or Clothing and Recombining Hakpaks

Lisa's Clothing Package is a relatively large file (over 100 MB) because it contains hundreds of custom outfits. In order to cut down on the size of our module (and the hakpaks it needed to function), we decided to rip the specific content we wanted from Lisa's hakpak, and make a new, smaller hakpak. The process for doing this will vary depending on the types of content you are working with, but overall the steps should be quite similar.

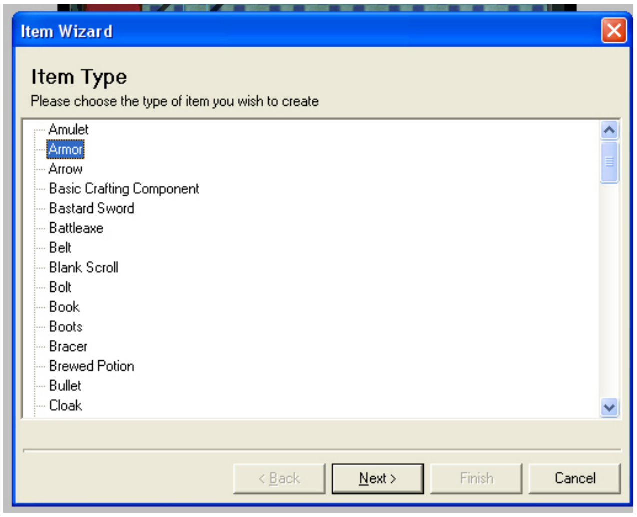
For this tutorial, I am going show you how to identify all the files associated with this dress:



Start by opening your module in the toolset, and create a blueprint for the object in question. In this case, I go to the Custom Item Wizard.



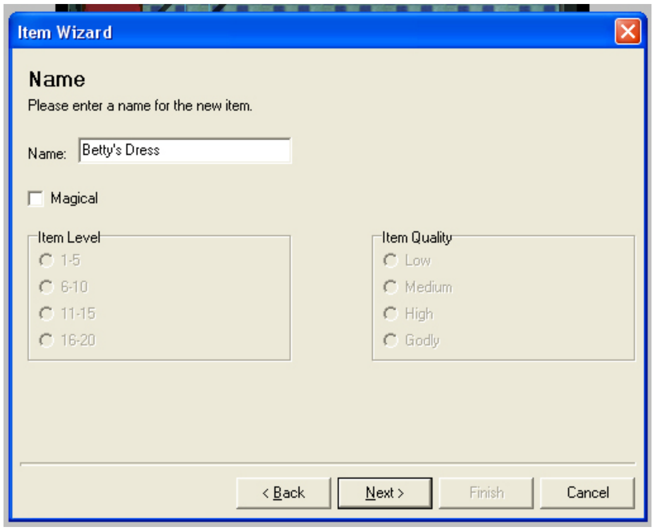
We want our item to be armor.



Item Type

Please choose the type of item you wish to create

Name your item.



Name

Please enter a name for the new item.

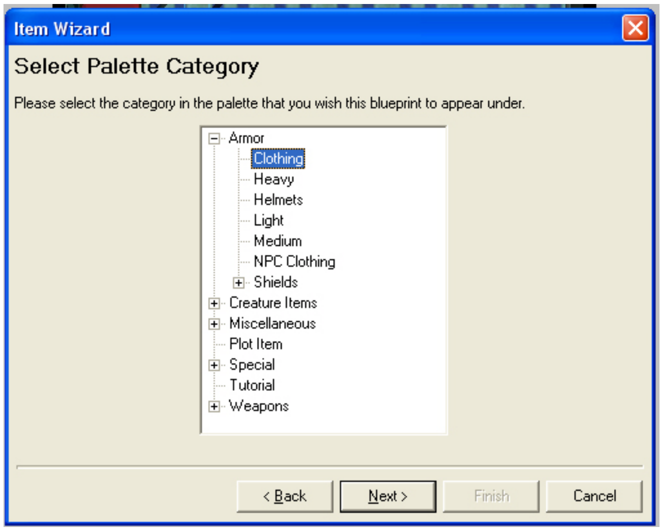
Name: Betty's Dress

□ Magical

Item Level
○ 1-5
○ 6-10
○ 11-15
○ 16-20

Item Quality
○ Low
○ Medium
○ High
○ Godly

Choose the category you want it to show up under.

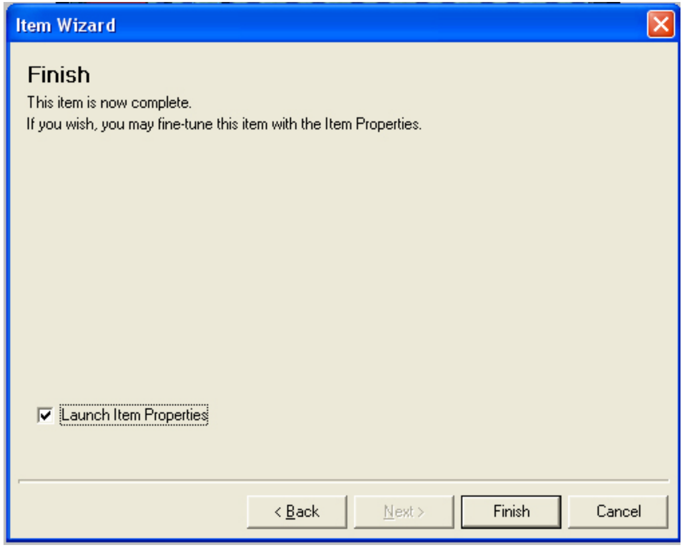


Select Palette Category

Please select the category in the palette that you wish this blueprint to appear under.

- Armor
  - Clothing
  - Heavy
  - Helmets
  - Light
  - Medium
  - NPC Clothing
  - Shields
- Creature Items
- Miscellaneous
- Plot Item
- Special
- Tutorial
- Weapons

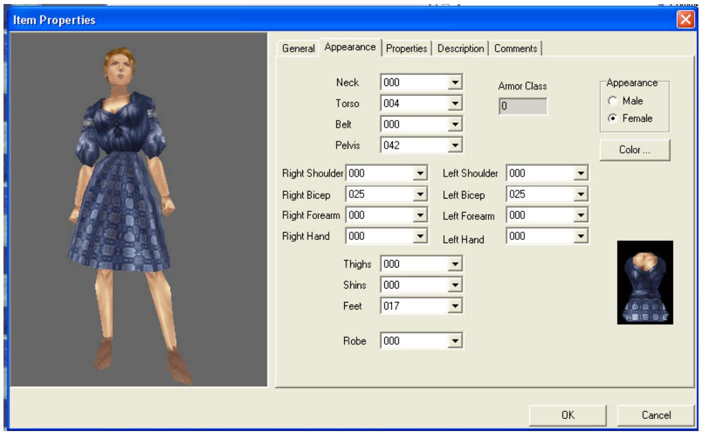
Now go into the item properties to configure how it.



Finish

This item is now complete.
If you wish, you may fine-tune this item with the Item Properties.

☑ Launch Item Properties



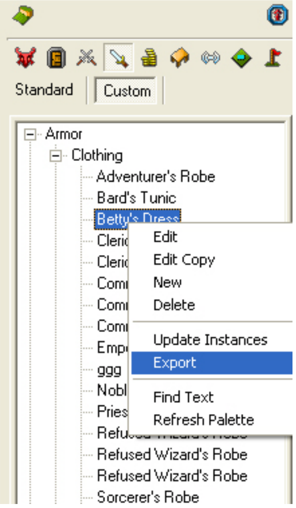
Neck	000		
Torso	004		
Belt	000		
Pelvis	042		
Right Shoulder	000	Left Shoulder	000
Right Bicep	025	Left Bicep	025
Right Forearm	000	Left Forearm	000
Right Hand	000	Left Hand	000
Thighs	000		
Shins	000		
Feet	017		
Robe	000		

Armor Class 0

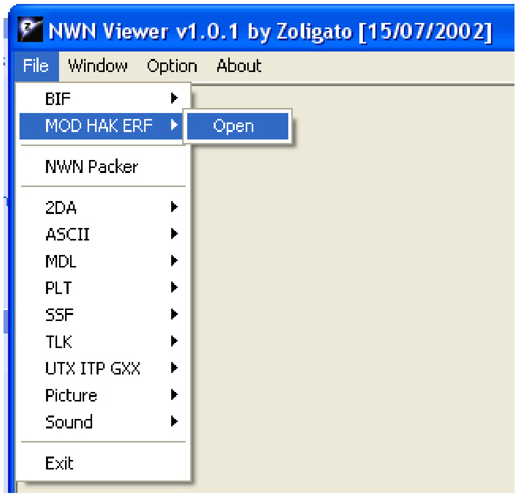
Appearance
○ Male
● Female

Take note of which custom pieces you are using. Here we have custom torso, pelvis, right and left biceps, and feet.

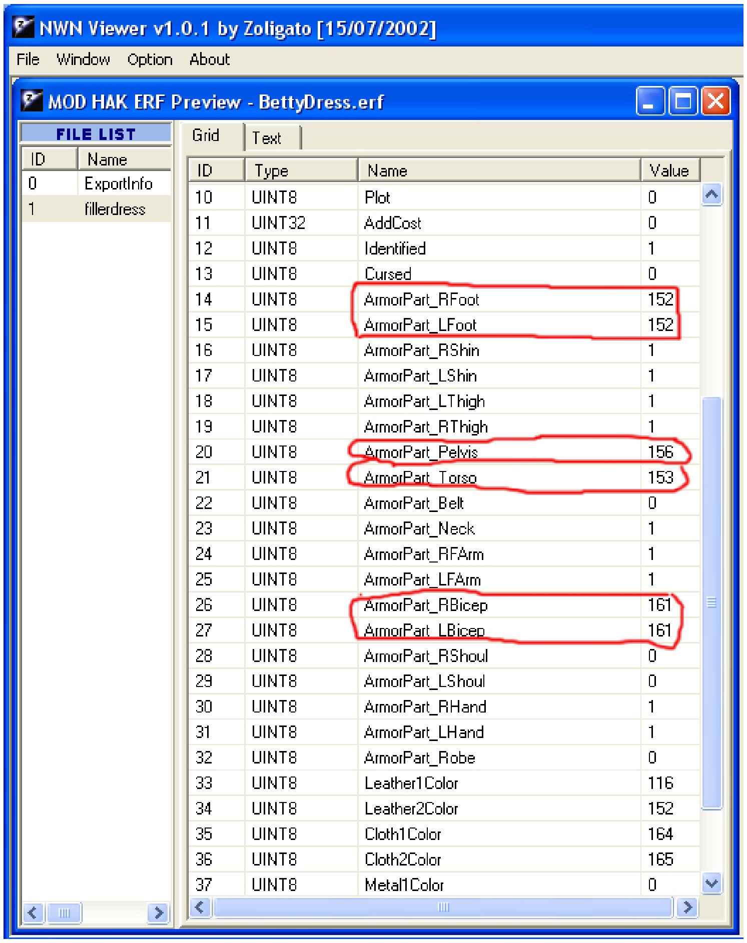
Now export this blueprint and save it as a .ERF.



Open the .ERF with Zoligato's image viewer. You may need to exit out of the *Neverwinter Nights* toolset before this program will run properly.



Choose the dress from the left column, and scroll through the right column to find the custom armor information. Make a note of these numbers, because they correspond to the object names within the hakpak.

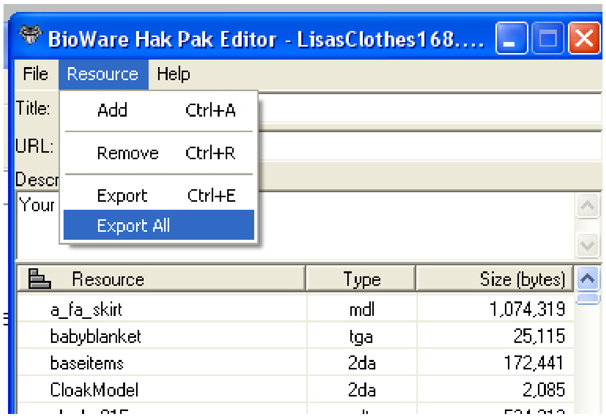


NWN Viewer v1.0.1 by Zoligato [15/07/2002]
File
Window
Option
About
MOD HAK ERF Preview - BettyDress.erf
FILE LIST
Grid
Text

ID	Name
0	ExportInfo
1	fillerdress

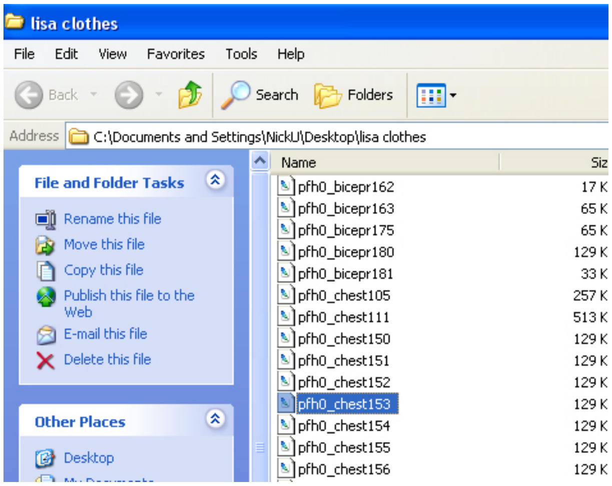
ID	Type	Name	Value
10	UINT8	Plot	0
11	UINT32	AddCost	0
12	UINT8	Identified	1
13	UINT8	Cursed	0
14	UINT8	ArmorPart_RFoot	152
15	UINT8	ArmorPart_LFoot	152
16	UINT8	ArmorPart_RShin	1
17	UINT8	ArmorPart_LShin	1
18	UINT8	ArmorPart_LThigh	1
19	UINT8	ArmorPart_RThigh	1
20	UINT8	ArmorPart_Pelvis	156
21	UINT8	ArmorPart_Torso	153
22	UINT8	ArmorPart_Belt	0
23	UINT8	ArmorPart_Neck	1
24	UINT8	ArmorPart_RFArm	1
25	UINT8	ArmorPart_LFArm	1
26	UINT8	ArmorPart_RBicep	161
27	UINT8	ArmorPart_LBicep	161
28	UINT8	ArmorPart_RShoul	0
29	UINT8	ArmorPart_LShoul	0
30	UINT8	ArmorPart_RHand	1
31	UINT8	ArmorPart_LHand	1
32	UINT8	ArmorPart_Robe	0
33	UINT8	Leather1Color	116
34	UINT8	Leather2Color	152
35	UINT8	Cloth1Color	164
36	UINT8	Cloth2Color	165
37	UINT8	Metal1Color	0

Now that we know what files we are looking for, open the hakpak with Bioware's Hakpak Editor. Export all of the files into a new folder to make searching through them easier.



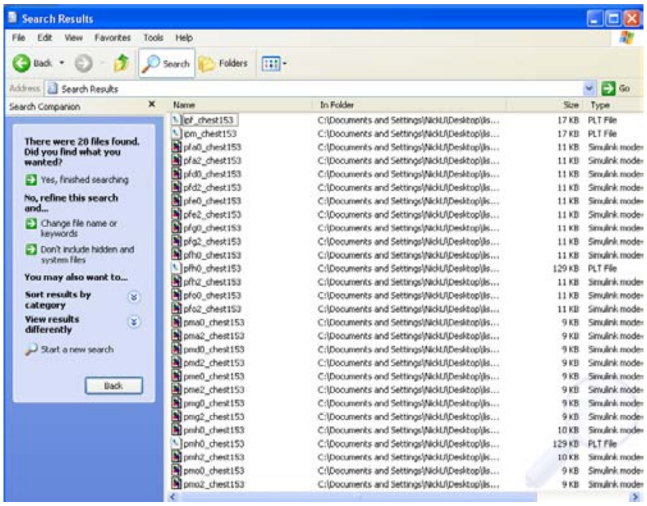
Resource	Type	Size (bytes)
a_fa_skirt	mdl	1,074,319
babyblanket	tga	25,115
baseitems	2da	172,441
CloakModel	2da	2,085

Let's find all the files related to torso piece #153. Keep in mind that naming conventions may vary slightly depending on who makes the hakpak.



Name	Siz
pfh0_bicepr162	17 K
pfh0_bicepr163	65 K
pfh0_bicepr175	65 K
pfh0_bicepr180	129 K
pfh0_bicepr181	33 K
pfh0_chest105	257 K
pfh0_chest111	513 K
pfh0_chest150	129 K
pfh0_chest151	129 K
pfh0_chest152	129 K
pfh0_chest153	129 K
pfh0_chest154	129 K
pfh0_chest155	129 K
pfh0_chest156	129 K

We now need to search for all the pieces with chest153 in their name.



There were 20 files found. Did you find what you wanted?

Name	In Folder	Size	Type
ipf_chest153	C:\Documents and Settings\NickU\Desktop\lis...	17 KB	PLT File
ipm_chest153	C:\Documents and Settings\NickU\Desktop\lis...	17 KB	PLT File
pfa0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfa2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfd0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfd2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfe0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfe2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfg0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfg2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfh0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfh0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	129 KB	PLT File
pfh2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfo0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pfo2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	11 KB	Simulink mode
pma0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode
pma2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode
pmd0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode
pmd2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode
pme0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode
pme2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode
pmg0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode
pmg2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode
pmh0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	10 KB	Simulink mode
pmh0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	129 KB	PLT File
pmh2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	10 KB	Simulink mode
pmo0_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode
pmo2_chest153	C:\Documents and Settings\NickU\Desktop\lis...	9 KB	Simulink mode

You now know what files you need in order for the torso section of the dress to show up properly in your module. Copy all of these files into the override folder (or a hakpak of your own).

Repeat this process for each piece of the armor that uses custom content.

As a final note, if you are really concerned with file sizes, it is important to learn how to read the four character prefix that these files have. Each combination corresponds to a specific race/sex/body type combination within *Neverwinter Nights*. If you do not need every single possible combination, you can cut out some files and save space.

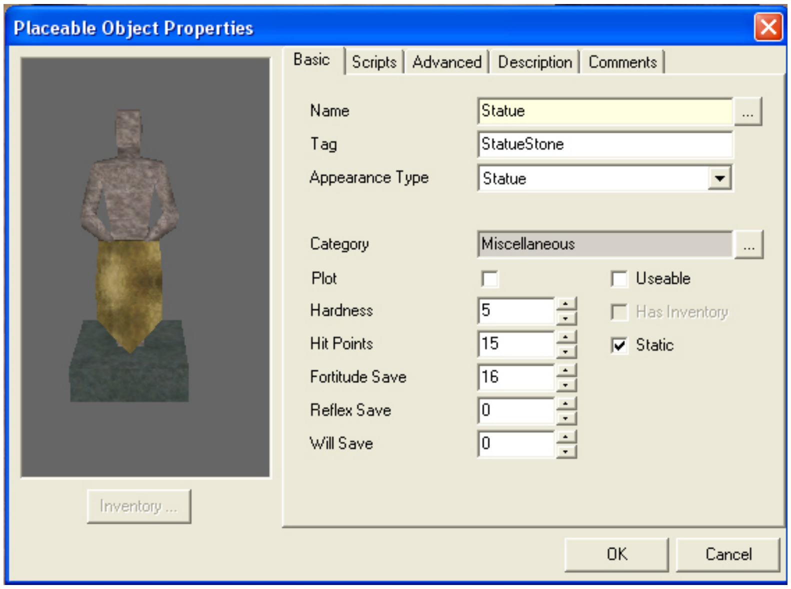
For example, I was only concerned with making this dress show up on one character, a female human of the normal body type. The prefix that corresponds to this combination is pfh0, so I just grabbed all the files with this prefix to save space. Also grab files such as ipf_chest153, because this is a general .PLT file that is needed for each version of torso 153.

Changing a Placeable's Model

This is a basic tutorial on using 3DS Max to create a new model for a placeable. For a more in depth look, please download Alex Swartz's video tutorial at http://gtproductions.net.

First, we decide what placeable we want to create a new model for. Keep in mind that we will need to find it in both the *Neverwinter Nights* toolset, as well as in NWN Explorer, so it is important to familiarize yourself with the layout of both of them.

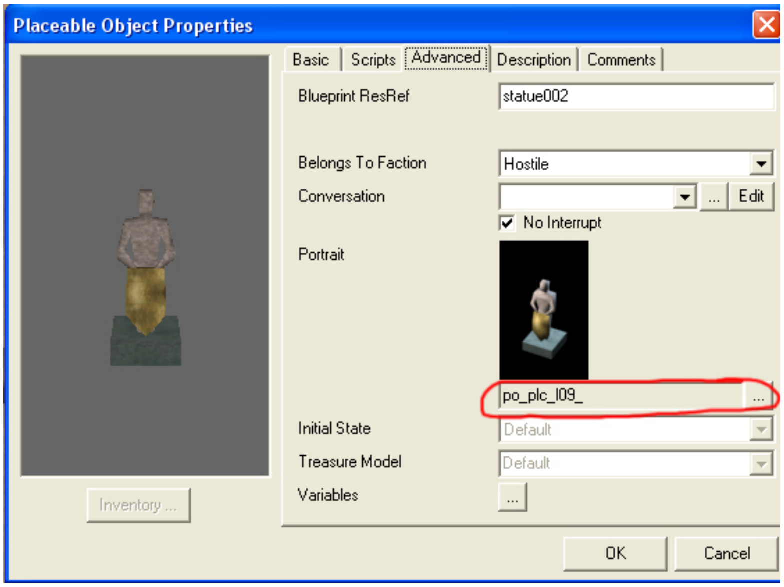
Start by finding the placeable you want to edit in the toolset. I am going to modify this statue:



Placeable Object Properties

Name	Statue		
Tag	StatueStone		
Appearance Type	Statue		
Category	Miscellaneous		
Plot	□	□ Useable	
Hardness	5	□ Has Inventory	
Hit Points	15	☑ Static	
Fortitude Save	16		
Reflex Save	0		
Will Save	0		

If you go to the Advanced tab, you can sometimes get information about what the file is named in the engine.

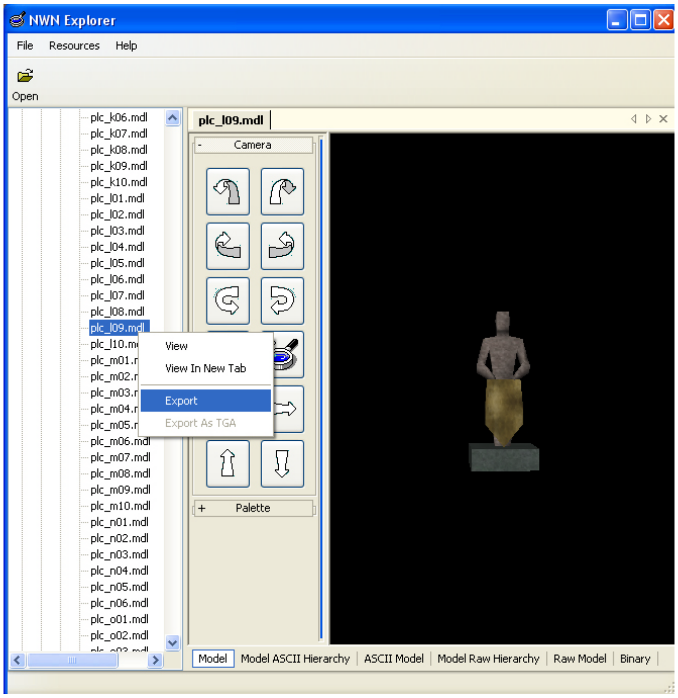


Placeable Object Properties

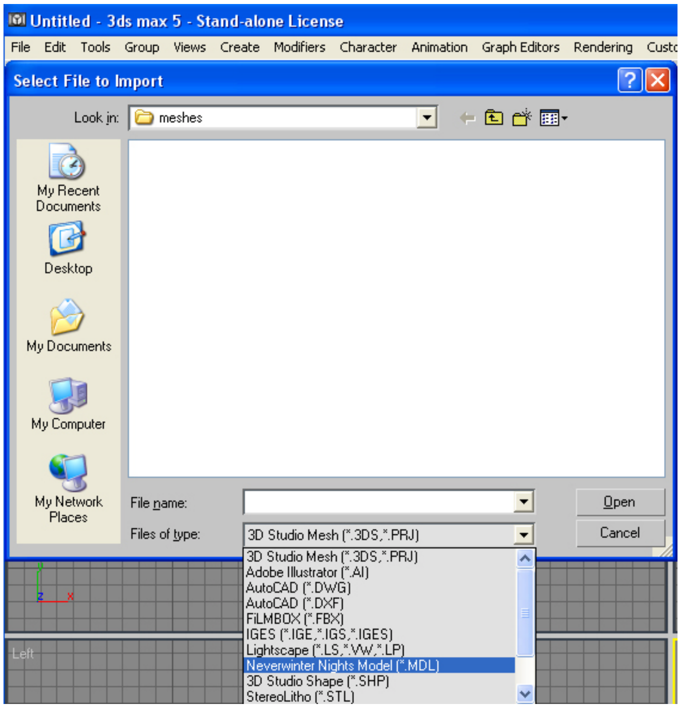
Blueprint ResRef	statue002
Belongs To Faction	Hostile
Conversation	
	☑ No Interrupt
Portrait	po_plc_l09_
Initial State	Default
Treasure Model	Default
Variables	...

This is actually the name of the portrait, but we can usually assume that the model is named something similar (in this case plc_l09).

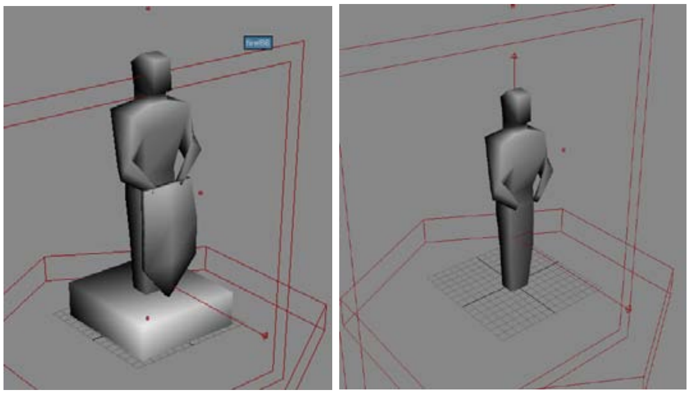
Now that that is decided, open *Neverwinter Nights* Explorer and use it to find and extract this model.



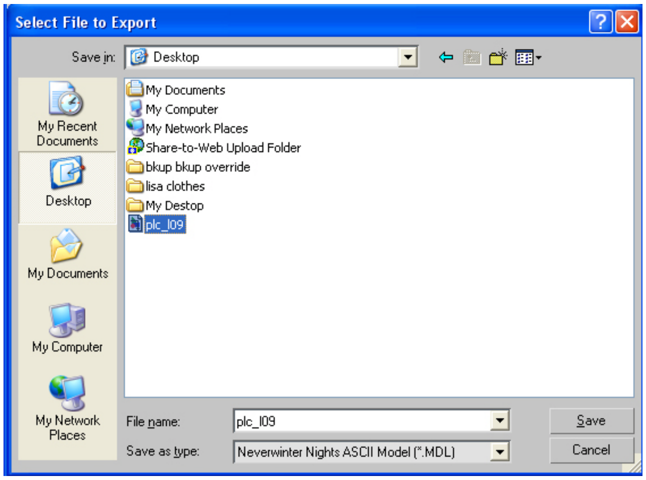
Now open up 3DS Max. As long as you have MDL Plug-In Suite 1.1 installed, you should have an option to import *Neverwinter Nights* .MDL files.



Now just edit the object. I took away the shield and the base pedestal.



Now export it as a *Neverwinter Nights* MDL. We want to overwrite the original file, so save it with the same name as the original.



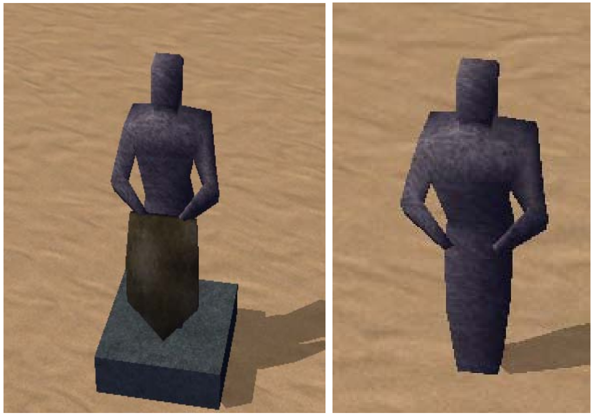
Select File to Export

Save in: Desktop

File name: plc_l09

Save as type: Neverwinter Nights ASCII Model (*.MDL)

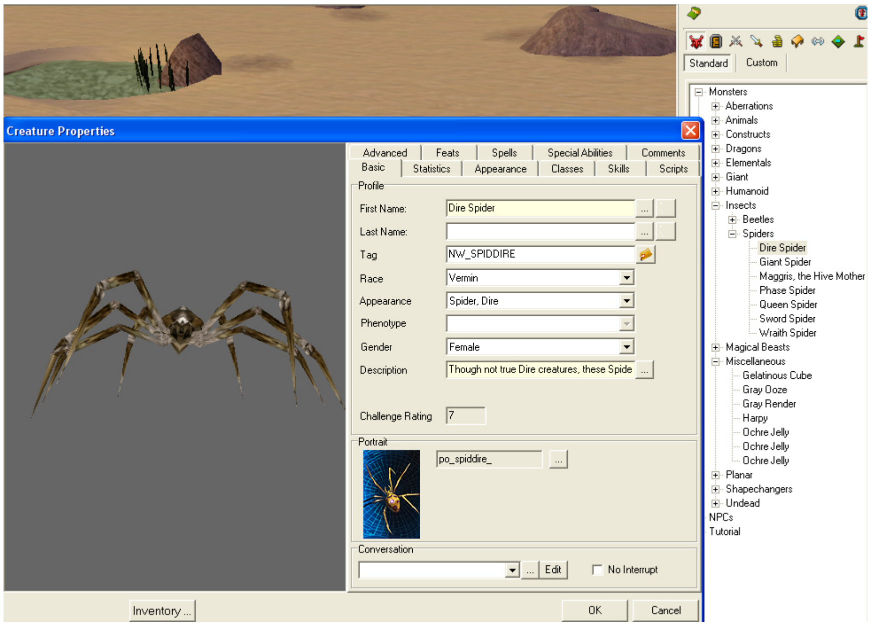
Now all you need to do is out this file into the override folder or a hakpak, and your customized model will show up instead of the original.



However, in many cases you will not want to simply replace an existing model. In order to add in a new model without overwriting it, you will need to edit the .2DA files, which I will cover later.

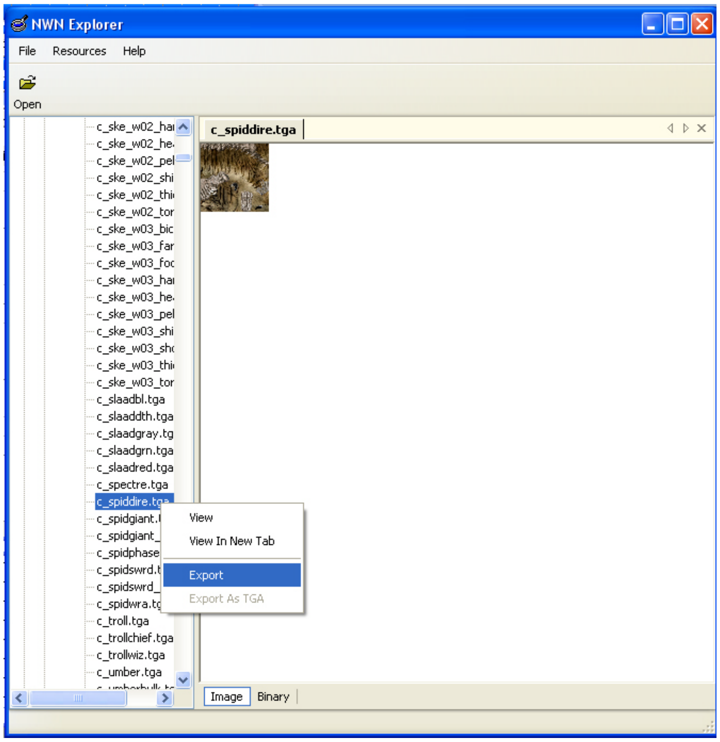
Recoloring a Monster

In this tutorial, we are going to find the texture files in *Neverwinter Nights* Editor and modify them, to create a new variation of an existing creature. Start by deciding what creature you want to modify, and find it in the *Neverwinter Nights* toolset. I am going to modify a Dire Spider.

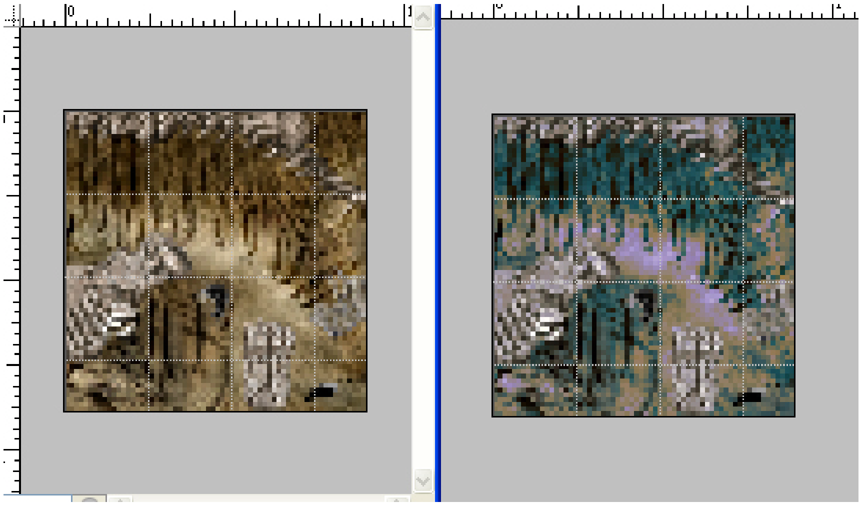


First Name: Dire Spider
Last Name:
Tag: NW_SPIDDIRE
Race: Vermin
Appearance: Spider, Dire
Phenotype:
Gender: Female
Description: Though not true Dire creatures, these Spide
Challenge Rating: 7
Portrait: po_spiddire_

Now you need to find the texture file in *Neverwinter Nights* Explorer and export it.

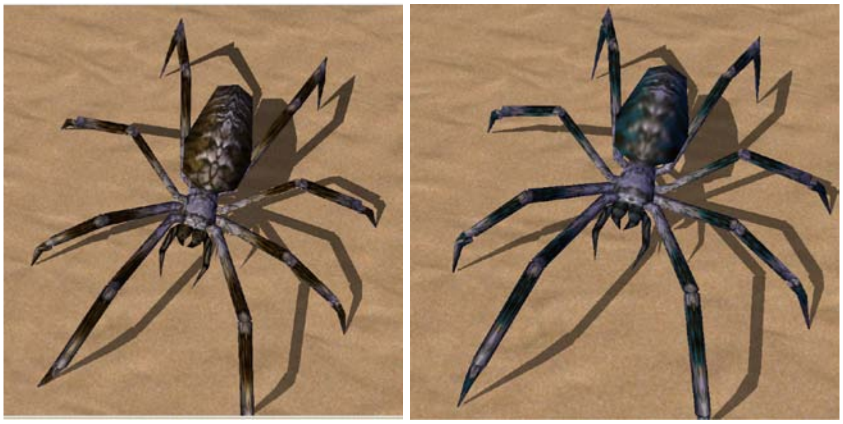


Open the file in Photoshop and modify it. I did some simple modifications using the replace color adjustment.



Now save this as a Targa file. We want to overwrite the original file, so save it with the same name as the original.

Now all you need to do is put this file into the override folder or a hakpak, and your customized monster will show up instead of the original.



It is important to note that this process overwrites the original creature. If you want to add in an alternate appearance for a creature, without overwriting the original, you will need to edit the .2DA files, which I will cover later.

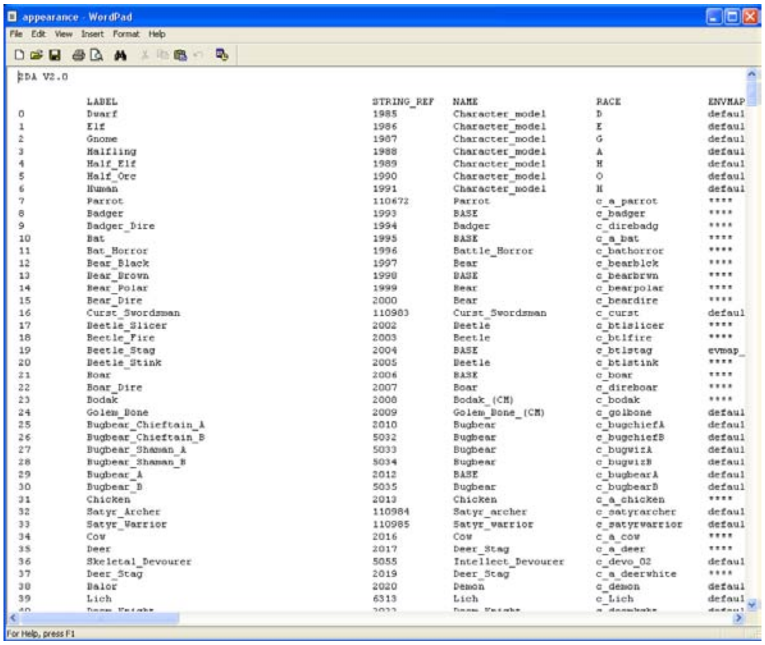
Making New Content That Does Not Overwrite Existing Content Via .2DA Editing

.2DA files are a sort of data file *Neverwinter Nights* uses. There are dozens of .2DA files in *Neverwinter Nights*. The ones I will be talking about keep track of what model file goes with a specific object within the game. We will be editing these files to add new models into the game, without overwriting the original files.

Start by downloading the source .2DA files from Bioware's *Neverwinter Nights* website. Do not use *Neverwinter Nights* Explorer to extract .2DA files, because *Neverwinter Nights* Explorer will give you the .2DA files as they were when the second expansion was released. Be sure that you are working off of the current files! If you work off of older versions, you will get the "invisible content" effect. Basically, if you start to notice that many standard *Neverwinter Nights* creatures or placeables are no longer showing up in your module, you have probably been working off of an outdated version.

Creatures and the Appearance.2DA File

In order to add a new creature into a module without overwriting a preexisting one, you need to modify the appearance.2DA file. Open it in Notepad or some other .txt editor.



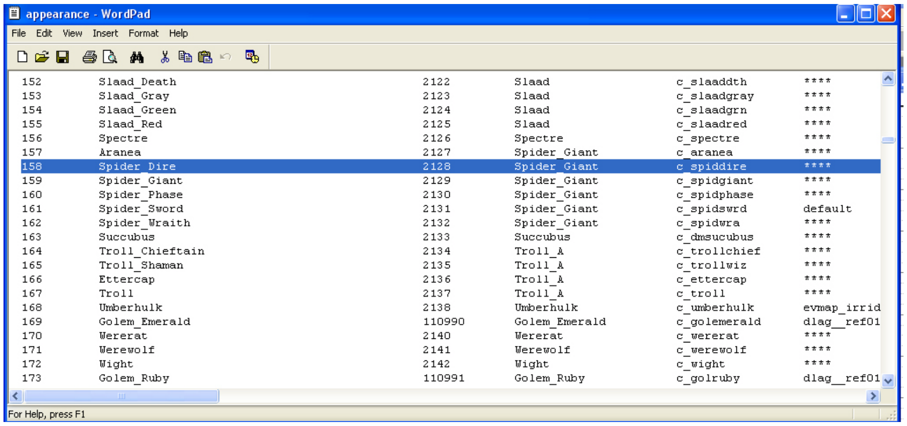
```
2DA V2.0

          LABEL                        STRING_REF  NAME                 RACE              ENVMAP
0         Dwarf                        1985        Character_model      D                 defaul
1         Elf                          1986        Character_model      E                 defaul
2         Gnome                        1987        Character_model      G                 defaul
3         Halfling                     1988        Character_model      A                 defaul
4         Half_Elf                     1989        Character_model      H                 defaul
5         Half_Orc                     1990        Character_model      O                 defaul
6         Human                        1991        Character_model      H                 defaul
7         Parrot                       110672      Parrot               c_a_parrot        ****
8         Badger                       1993        BASE                 c_badger          ****
9         Badger_Dire                  1994        Badger               c_direbadg        ****
10        Bat                          1995        BASE                 c_a_bat           ****
11        Bat_Horror                   1996        Battle_Horror        c_bathorror       ****
12        Bear_Black                   1997        Bear                 c_bearblck        ****
13        Bear_Brown                   1998        BASE                 c_bearbrwn        ****
14        Bear_Polar                   1999        Bear                 c_bearpolar       ****
15        Bear_Dire                    2000        Bear                 c_beardire        ****
16        Curst_Swordsman              110983      Curst_Swordsman      c_curst           defaul
17        Beetle_Slicer                2002        Beetle               c_btlslicer       ****
18        Beetle_Fire                  2003        Beetle               c_btlfire         ****
19        Beetle_Stag                  2004        BASE                 c_btlstag         evmap_
20        Beetle_Stink                 2005        Beetle               c_btlstink        ****
21        Boar                         2006        BASE                 c_boar            ****
22        Boar_Dire                    2007        Boar                 c_direboar        ****
23        Bodak                        2008        Bodak_(CM)           c_bodak           ****
24        Golem_Bone                   2009        Golem_Bone_(CM)      c_golbone         defaul
25        Bugbear_Chieftain_A          2010        Bugbear              c_bugchiefA       defaul
26        Bugbear_Chieftain_B          5032        Bugbear              c_bugchiefB       defaul
27        Bugbear_Shaman_A             5033        Bugbear              c_bugwizA         defaul
28        Bugbear_Shaman_B             5034        Bugbear              c_bugwizB         defaul
29        Bugbear_A                    2012        BASE                 c_bugbearA        defaul
30        Bugbear_B                    5035        Bugbear              c_bugbearB        defaul
31        Chicken                      2013        Chicken              c_a_chicken       ****
32        Satyr_Archer                 110984      Satyr_archer         c_satyrarcher     defaul
33        Satyr_Warrior                110985      Satyr_warrior        c_satyrwarrior    defaul
34        Cow                          2016        Cow                  c_a_cow           ****
35        Deer                         2017        Deer_Stag            c_a_deer          ****
36        Skeletal_Devourer            5055        Intellect_Devourer   c_devo_02         defaul
37        Deer_Stag                    2019        Deer_Stag            c_a_deerwhite     ****
38        Balor                        2020        Demon                c_demon           defaul
39        Lich                         6313        Lich                 c_Lich            defaul
```

There is a lot of information in this file, but I am only going to cover what you need to know to get your new creature into the engine. For a much more detailed explanation, take a look at the *Neverwinter Nights* Custom Content Guide.

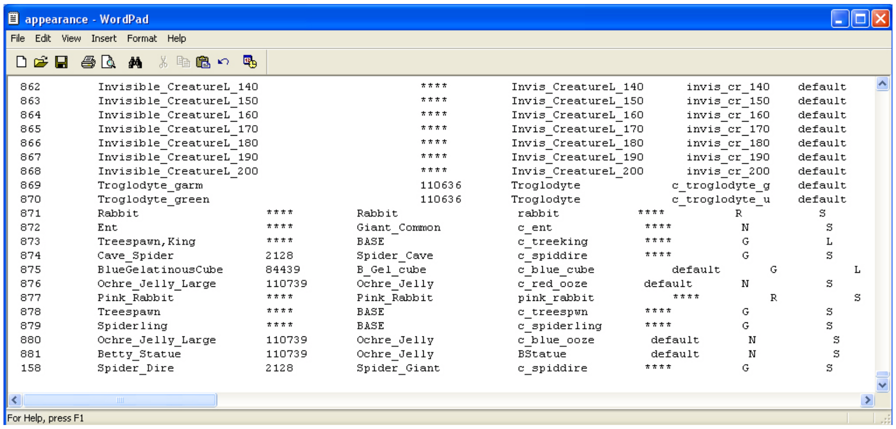
Each line corresponds to one creature in the engine. Basically, you are going to copy and paste an entire line, and update the important pieces for your specific monster. I am going to get the custom Dire Spider I made earlier into the engine without overwriting the original.

Find the line corresponding to the creature that most resembles the new one you will be putting in. If you only did a simple texture modification, you simply choose the creature that you modified.



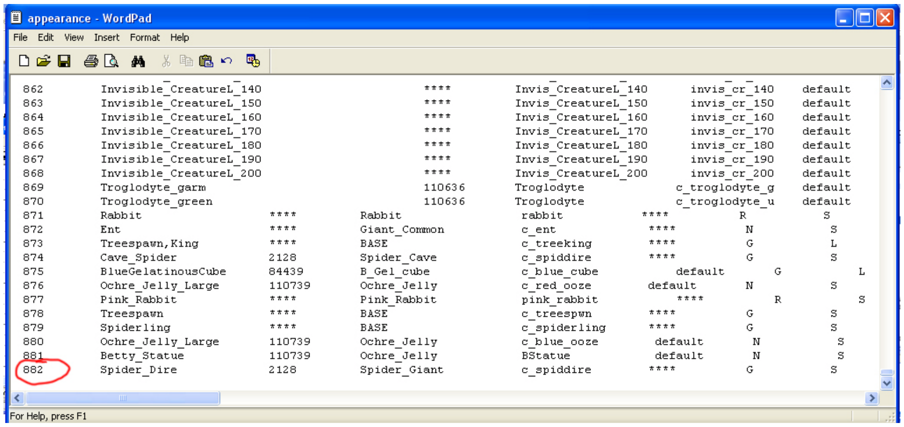
```
152        Slaad_Death                         2122      Slaad              c_slaaddth     ****
153        Slaad_Gray                          2123      Slaad              c_slaadgray    ****
154        Slaad_Green                         2124      Slaad              c_slaadgrn     ****
155        Slaad_Red                           2125      Slaad              c_slaadred     ****
156        Spectre                             2126      Spectre            c_spectre      ****
157        Aranea                              2127      Spider_Giant       c_aranea       ****
158        Spider_Dire                         2128      Spider_Giant       c_spiddire     ****
159        Spider_Giant                        2129      Spider_Giant       c_spidgiant    ****
160        Spider_Phase                        2130      Spider_Giant       c_spidphase    ****
161        Spider_Sword                        2131      Spider_Giant       c_spidswrd     default
162        Spider_Wraith                       2132      Spider_Giant       c_spidwra      ****
163        Succubus                            2133      Succubus           c_dmsucubus    ****
164        Troll_Chieftain                     2134      Troll_A            c_trollchief   ****
165        Troll_Shaman                        2135      Troll_A            c_trollwiz     ****
166        Ettercap                            2136      Troll_A            c_ettercap     ****
167        Troll                               2137      Troll_A            c_troll        ****
168        Umberhulk                           2138      Umberhulk          c_umberhulk    evmap_irrid
169        Golem_Emerald                       110990    Golem_Emerald      c_golemerald   dlag__ref01
170        Wererat                             2140      Wererat            c_wererat      ****
171        Werewolf                            2141      Werewolf           c_werewolf     ****
172        Wight                               2142      Wight              c_wight        ****
173        Golem_Ruby                          110991    Golem_Ruby         c_golruby      dlag__ref01
```

Paste this line at the end of the file.



```
862        Invisible_CreatureL_140                     ****      Invis_CreatureL_140    invis_cr_140   default
863        Invisible_CreatureL_150                     ****      Invis_CreatureL_150    invis_cr_150   default
864        Invisible_CreatureL_160                     ****      Invis_CreatureL_160    invis_cr_160   default
865        Invisible_CreatureL_170                     ****      Invis_CreatureL_170    invis_cr_170   default
866        Invisible_CreatureL_180                     ****      Invis_CreatureL_180    invis_cr_180   default
867        Invisible_CreatureL_190                     ****      Invis_CreatureL_190    invis_cr_190   default
868        Invisible_CreatureL_200                     ****      Invis_CreatureL_200    invis_cr_200   default
869        Troglodyte_garm                             110636    Troglodyte          c_troglodyte_g    default
870        Troglodyte_green                            110636    Troglodyte          c_troglodyte_u    default
871        Rabbit                 ****      Rabbit                rabbit        ****         R           S
872        Ent                    ****      Giant_Common          c_ent          ****        N           S
873        Treespawn,King         ****      BASE                  c_treeking     ****        G           L
874        Cave_Spider            2128      Spider_Cave           c_spiddire     ****        G           S
875        BlueGelatinousCube     84439     B_Gel_cube            c_blue_cube       default       G           L
876        Ochre_Jelly_Large      110739    Ochre_Jelly           c_red_ooze    default      N           S
877        Pink_Rabbit            ****      Pink_Rabbit           pink_rabbit       ****          R           S
878        Treespawn              ****      BASE                  c_treespwn     ****        G           S
879        Spiderling             ****      BASE                  c_spiderling   ****        G           S
880        Ochre_Jelly_Large      110739    Ochre_Jelly           c_blue_ooze     default     N           S
881        Betty_Statue           110739    Ochre_Jelly           BStatue         default     N           S
158        Spider_Dire            2128      Spider_Giant          c_spiddire     ****        G           S
```

Start by updating the ID number.

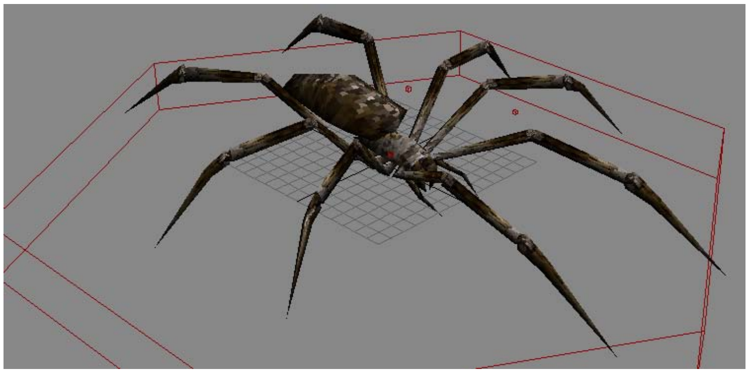


```
862        Invisible_CreatureL_140                     ****      Invis_CreatureL_140    invis_cr_140   default
863        Invisible_CreatureL_150                     ****      Invis_CreatureL_150    invis_cr_150   default
864        Invisible_CreatureL_160                     ****      Invis_CreatureL_160    invis_cr_160   default
865        Invisible_CreatureL_170                     ****      Invis_CreatureL_170    invis_cr_170   default
866        Invisible_CreatureL_180                     ****      Invis_CreatureL_180    invis_cr_180   default
867        Invisible_CreatureL_190                     ****      Invis_CreatureL_190    invis_cr_190   default
868        Invisible_CreatureL_200                     ****      Invis_CreatureL_200    invis_cr_200   default
869        Troglodyte_garm                             110636    Troglodyte          c_troglodyte_g    default
870        Troglodyte_green                            110636    Troglodyte          c_troglodyte_u    default
871        Rabbit                 ****      Rabbit                rabbit        ****         R           S
872        Ent                    ****      Giant_Common          c_ent          ****        N           S
873        Treespawn,King         ****      BASE                  c_treeking     ****        G           L
874        Cave_Spider            2128      Spider_Cave           c_spiddire     ****        G           S
875        BlueGelatinousCube     84439     B_Gel_cube            c_blue_cube       default       G           L
876        Ochre_Jelly_Large      110739    Ochre_Jelly           c_red_ooze    default      N           S
877        Pink_Rabbit            ****      Pink_Rabbit           pink_rabbit       ****          R           S
878        Treespawn              ****      BASE                  c_treespwn     ****        G           S
879        Spiderling             ****      BASE                  c_spiderling   ****        G           S
880        Ochre_Jelly_Large      110739    Ochre_Jelly           c_blue_ooze     default     N           S
881        Betty_Statue           110739    Ochre_Jelly           BStatue         default     N           S
882        Spider_Dire            2128      Spider_Giant          c_spiddire     ****        G           S
```

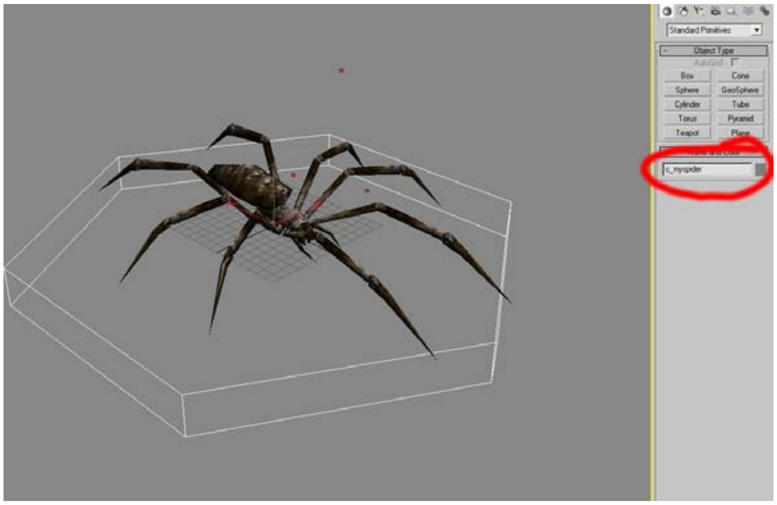
There is one other piece of text we need to update. The fifth column tells the engine what the specific .MDL file is named that contains the creature. We need to get "c_spiddire.MDL" from *Neverwinter Nights* Explorer and open it in 3DS Max. You should know how to do this by now.

You may recall that the spider texture that we edited earlier was called "c_spiddire.TGA." Standard naming convention says that the .MDL and .TGA files are always named the same thing. Keep that in mind as you edit the model.

Import the creature into 3DS Max. If the texture named “c_spiddire.TGA” is in the same directory as the model “c_spiddire.MDL,” then the creature will appear inn 3DS Max properly textured. However, this doesn’t really matter, because we are about to change the texture file anyway.

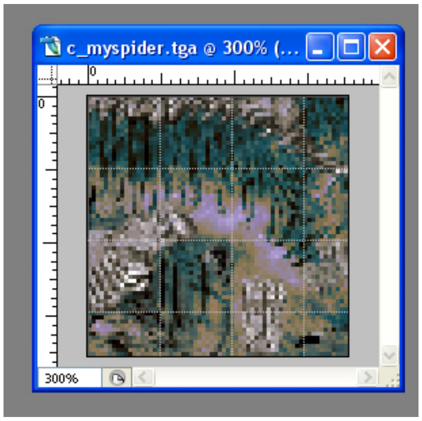


First, rename the creature to something unique. I am going to call it “c_myspider.” In 3DS Max, choose the red hexagonal base around the creature and rename it. Remember this name; we will be using it again later.



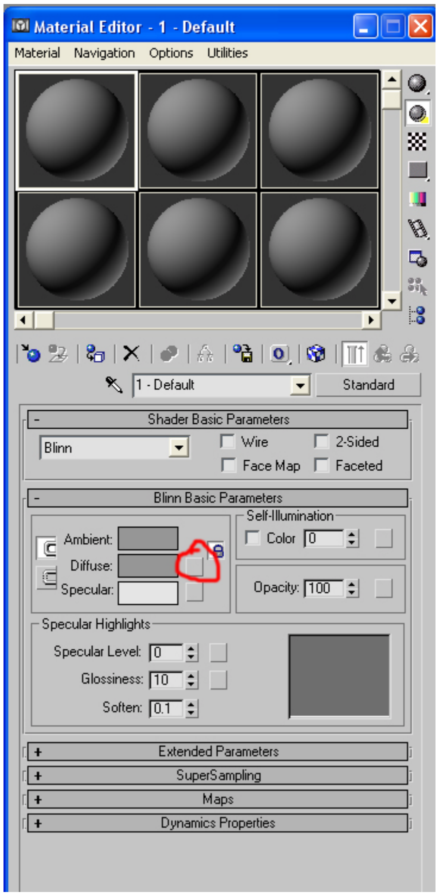
c_myspider

Update your custom texture so that it shares the same name as your custom model.



Now you need to assign this texture to the creature model in 3DS Max. You do not need to adjust any UV's because its geometry is exactly the same as the original spider.

Press "M" to open up the Material Editor. Click on the small gray box to the right of the Diffuse box.



1 - Default

Blinn

□ Wire □ 2-Sided

□ Face Map □ Faceted

Ambient:

Diffuse:

Specular:

□ Color 0

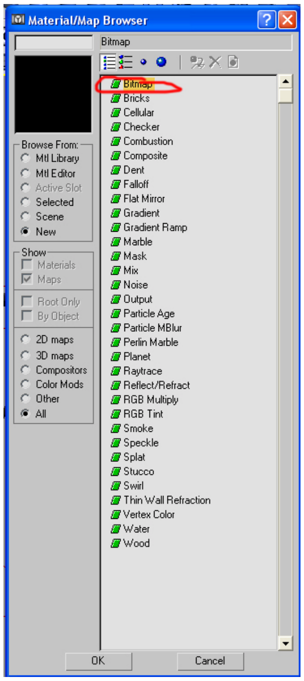
Opacity: 100

Specular Level: 0

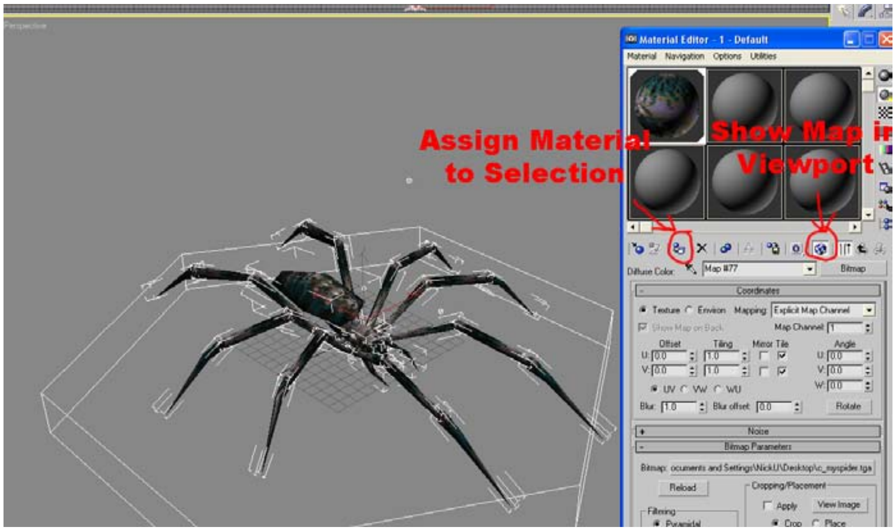
Glossiness: 10

Soften: 0.1

Double click on Bitmap, and then choose your custom texture.



Select all the objects in your scene by pressing Control-A, and then click the “Assign material to Selection” button. Make sure that the “Show Map in Viewport” button is active.



Diffuse Color: Map #77 Bitmap

● Texture ○ Environ Mapping: Explicit Map Channel

Map Channel: 1

Offset U: 0.0 V: 0.0; Tiling U: 1.0 V: 1.0; Mirror □ □; Tile ☑ ☑; Angle U: 0.0 V: 0.0 W: 0.0

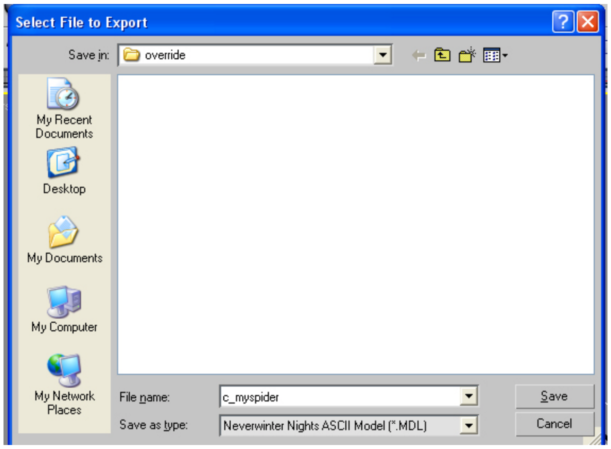
● UV ○ VW ○ WU

Blur: 1.0 Blur offset: 0.0

Bitmap: ocuments and Settings\NickU\Desktop\c_myspider.tga

□ Apply ● Crop ○ Place ● Pyramidal

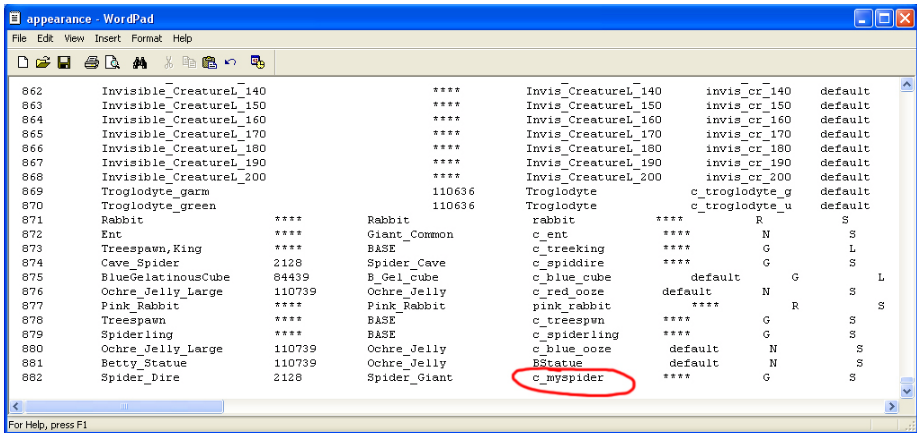
Now we can export the new creature. Be sure to export it as a *Neverwinter Nights* MDL with the same name that you gave the creature base earlier on.



File name: c_myspider

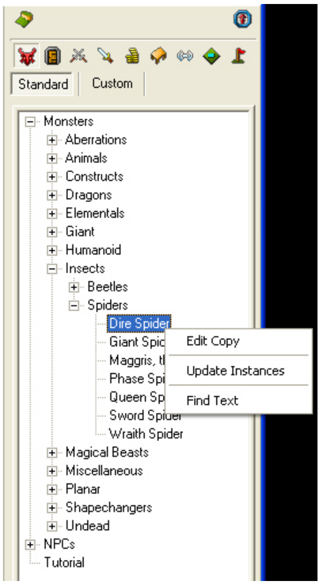
Save as type: Neverwinter Nights ASCII Model (*.MDL)

Now you need to finish editing the .2DA file. In the fifth column, change the default model name to the model name you have been using for your creature. Save the appearance.2DA file.

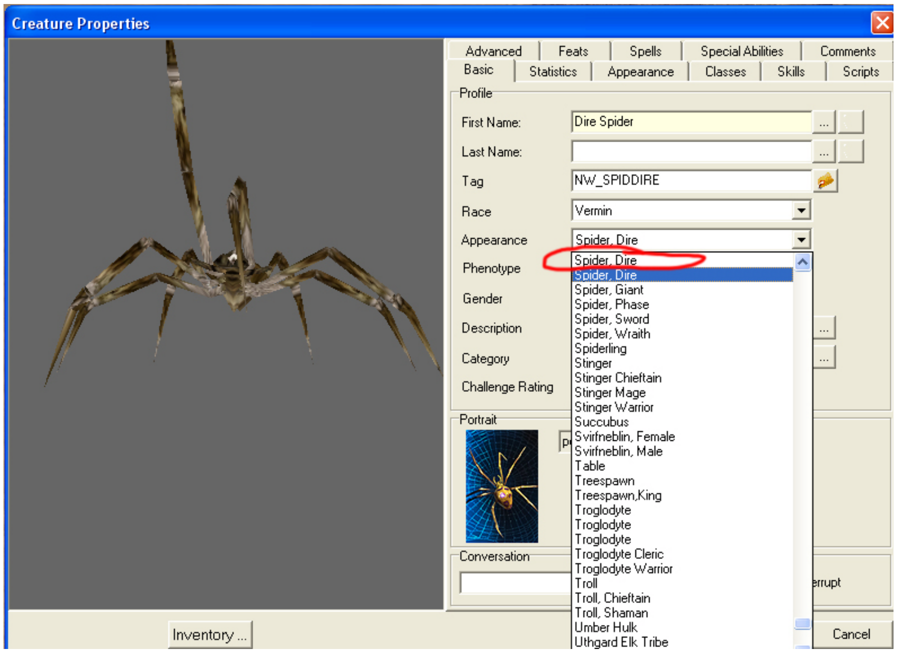


```
862      Invisible_CreatureL_140                          ****      Invis_CreatureL_140     invis_cr_140    default
863      Invisible_CreatureL_150                          ****      Invis_CreatureL_150     invis_cr_150    default
864      Invisible_CreatureL_160                          ****      Invis_CreatureL_160     invis_cr_160    default
865      Invisible_CreatureL_170                          ****      Invis_CreatureL_170     invis_cr_170    default
866      Invisible_CreatureL_180                          ****      Invis_CreatureL_180     invis_cr_180    default
867      Invisible_CreatureL_190                          ****      Invis_CreatureL_190     invis_cr_190    default
868      Invisible_CreatureL_200                          ****      Invis_CreatureL_200     invis_cr_200    default
869      Troglodyte_garm                                  110636    Troglodyte          c_troglodyte_g  default
870      Troglodyte_green                                 110636    Troglodyte          c_troglodyte_u  default
871      Rabbit                 ****       Rabbit          rabbit            ****       R         S
872      Ent                    ****       Giant_Common    c_ent              ****      N         S
873      Treespawn,King         ****       BASE            c_treeking         ****      G         L
874      Cave_Spider            2128       Spider_Cave     c_spiddire         ****      G         S
875      BlueGelatinousCube     84439      B_Gel_cube      c_blue_cube          default     G         L
876      Ochre_Jelly_Large      110739     Ochre_Jelly     c_red_ooze        default    N         S
877      Pink_Rabbit            ****       Pink_Rabbit     pink_rabbit           ****     R         S
878      Treespawn              ****       BASE            c_treespwn         ****      G         S
879      Spiderling             ****       BASE            c_spiderling       ****      G         S
880      Ochre_Jelly_Large      110739     Ochre_Jelly     c_blue_ooze        default    N         S
881      Betty_Statue           110739     Ochre_Jelly     BStatue            default    N         S
882      Spider_Dire            2128       Spider_Giant    c_myspider         ****      G         S
```

Now take the model, texture, and appearance.2da file and put them in the override folder or a hakpak. Open your module in the *Neverwinter Nights* toolset, and find the original creature you based your custom one off of in the creature palette. Right click and select Edit Copy.



You need to update the appearance dropdown menu, so that this creature loads the appropriate model. Since we left most of the original fields in the appearance.2DA file unchanged, your custom creature will show up with the same name as the original. Choose your custom monster.



First Name: Dire Spider

Last Name:

Tag: NW_SPIDDIRE

Race: Vermin

Appearance: Spider, Dire

Your monster is now in *Neverwinter Nights*, and you have not overwritten the original.

Placeables and the Placeables.2DA File

Getting a custom placeable into *Neverwinter Nights* is essentially the same as getting a custom creature into *Neverwinter Nights*. Instead of showing the entire process again, I'll quickly review it and point out the few differences. For more information regarding the process of creating and importing custom placeables, take a look at Alex Swartz's *Neverwinter Nights* videos at http://gtproductions.net/.

First of all, you will be working off of the placeables.2DA file. Like before, start by copying an entire line of a similar placeable, and pasting it at the bottom.

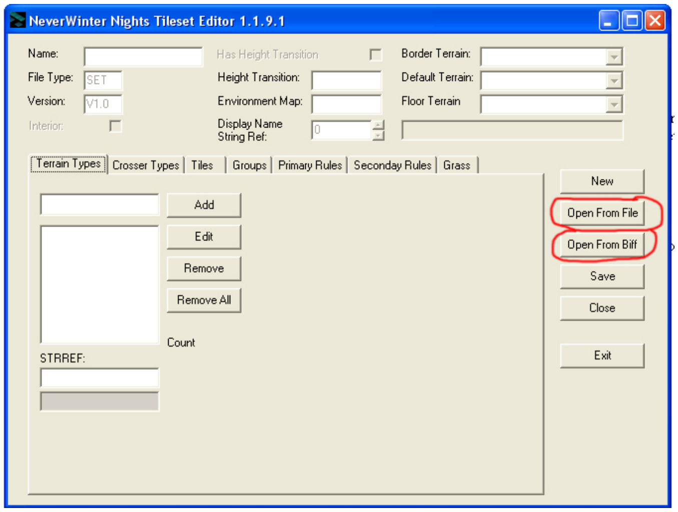
In 3DS Max, be sure the hexagonal base is given a unique object name. That name will also be used when you export your file, and when you update the line of text in the fourth column of the placeables.2DA file.

Editing the NWN Grass

Before you decide to change the settings of the *Neverwinter Nights* grass, it is important that you understand that the tool used for this is very buggy and may corrupt your tileset. If you decide to do this, be sure you have backups of all your files.

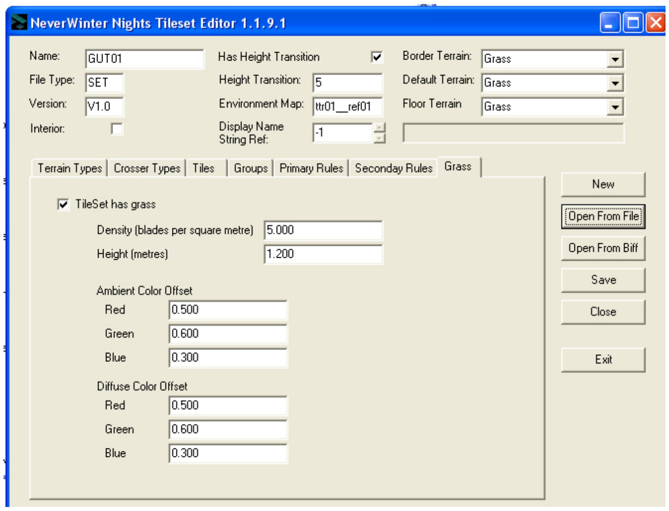
First you need to download a program called BuildTil. It is the official tileset editor released by Bioware.

If you want to edit an existing *Neverwinter Nights* tileset, select Open From Biff. If you want to edit a custom tileset, you need to select Open From File. If you are editing a custom tileset, you specifically need the .SET file, which you may first need to extract from the tileset's hakpak. I am going to change the grass color of a tileset I downloaded online.



File Type: SET
Version: V1.0
Display Name String Ref: 0

Click on the Grass tab. There you can adjust the density, height, and color of the grass.



Name: GUT01
File Type: SET
Version: V1.0
Interior: ☐
Has Height Transition ☑
Height Transition: 5
Environment Map: ttr01__ref01
Display Name String Ref: -1
Border Terrain: Grass
Default Terrain: Grass
Floor Terrain Grass

☑ TileSet has grass

Density (blades per square metre) 5.000
Height (metres) 1.200

Ambient Color Offset
Red 0.500
Green 0.600
Blue 0.300

Diffuse Color Offset
Red 0.500
Green 0.600
Blue 0.300

Save the file and throw it into the override folder, or a hakpak.

I just want to once again point out how unstable BuildTil is by mentioning that within 5 minutes of completing the steps outlined in this tutorial, my computer ran into several errors and needed to be restarted. Ask yourself if customized grass is really that important to your module before you go through this procedure.

A Quick Overview of Tileset Editing

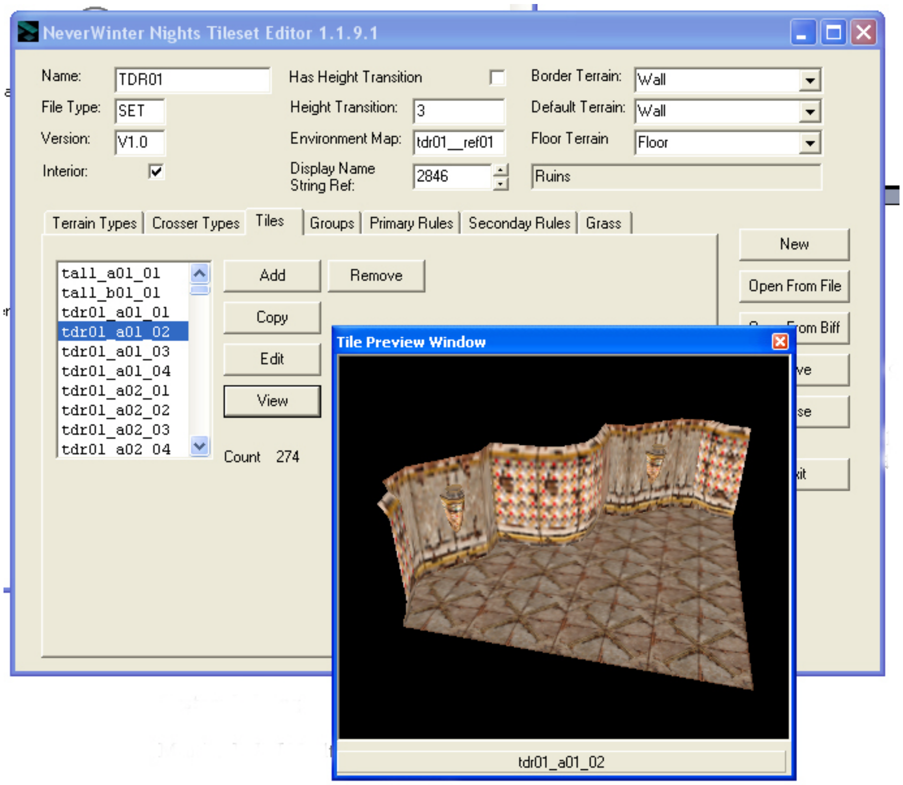
This is a quick overview of tileset editing in *Neverwinter Nights*. For a more complete look, see Alex Swartz's videos at http://gtproductions.net/.

Start by gathering all of the files from the tileset you want to edit. If it is one of the default *Neverwinter Nights* tilesets, you can use *Neverwinter Nights* Explorer to get them, and if it is a custom tileset, you can use Bioware's Hakpak editor to get them. If you are modifying a tileset that comes in a hakpak, when you change the files you should add the custom ones to the hakpak instead of adding them to the override folder.

If you want to change a tileset's texture, you can generally do so easily by editing the appropriate textures in Photoshop. However, some tilesets have many slight variations of the same texture, so you must change all of them accordingly.

If you want to change the geometry for a tileset, you need to import each tile individually into 3DS Max, make the changes, make sure the textures are still mapped correctly, and then export it.

If you are looking to modify a few specific tiles, you can use BuildTil as a sort of tile viewer. Go to the Tiles tab, scroll through the list of tiles, and click view.



NeverWinter Nights Tileset Editor 1.1.9.1

Name: TDR01
File Type: SET
Version: V1.0
Interior: ☑
Has Height Transition ☐
Height Transition: 3
Environment Map: tdr01__ref01
Display Name String Ref: 2846
Border Terrain: Wall
Default Terrain: Wall
Floor Terrain: Floor
Ruins

```
tall_a01_01
tall_b01_01
tdr01_a01_01
tdr01_a01_02
tdr01_a01_03
tdr01_a01_04
tdr01_a02_01
tdr01_a02_02
tdr01_a02_03
tdr01_a02_04
```

Count 274

Tile Preview Window

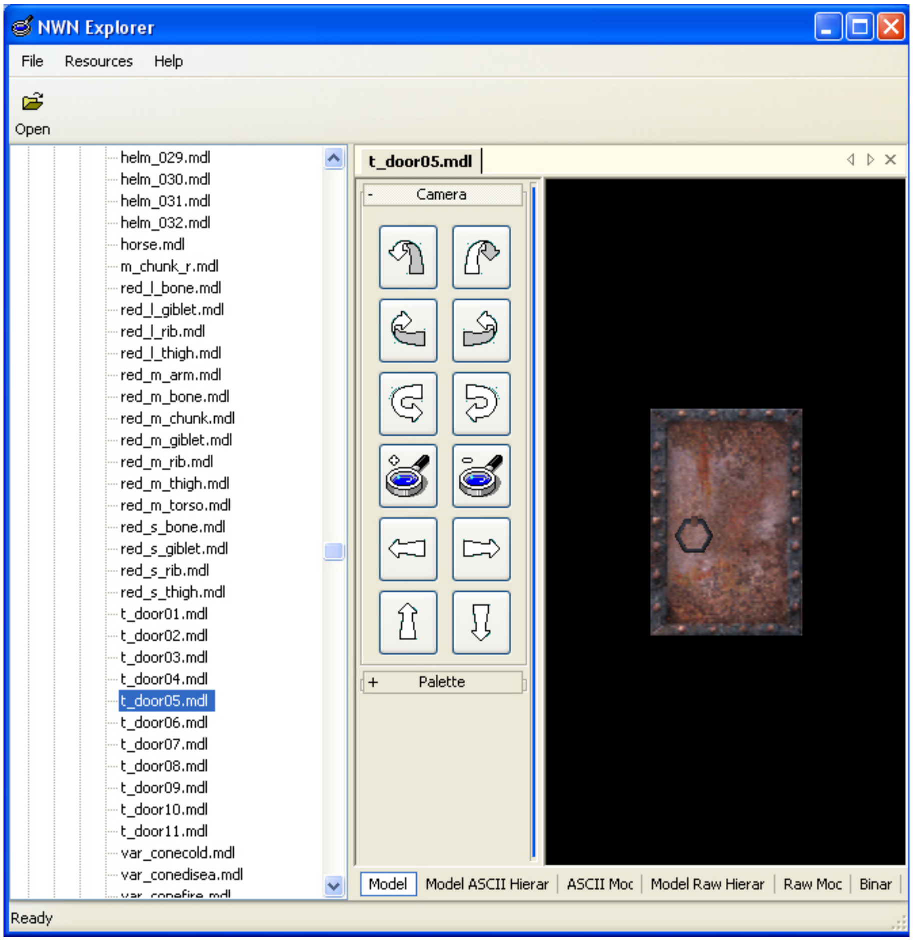
tdr01_a01_02

In general, BuildTil is a very buggy program. This is the only time I recommend actually using it.

Making a Custom Door

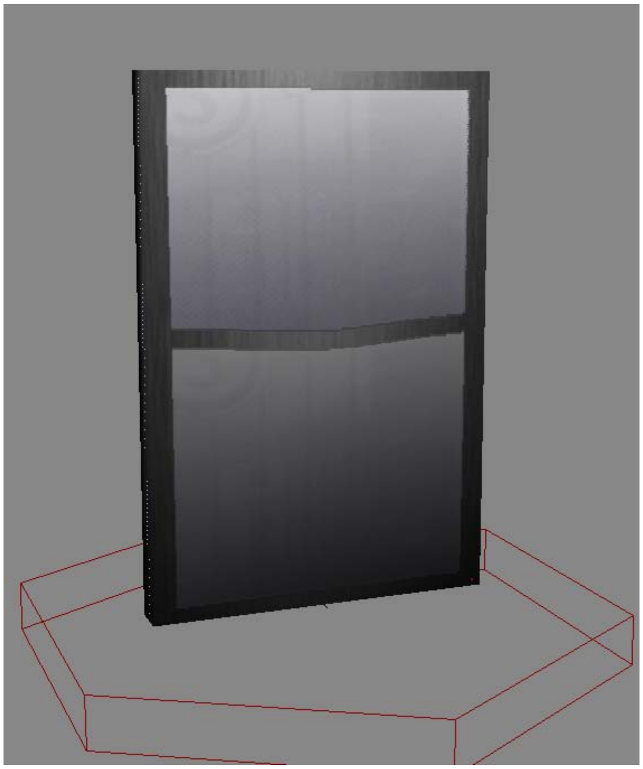
This will show you how to overwrite an existing *Neverwinter Nights* door, and put in one of your own creation. If you want to make a non overwriting door, you will need to edit the doortypes.2DA file.

There are a few different types of doors in *Neverwinter Nights*. Some open down the middle, some open upwards or downwards, and some have other animations. Find a door with the appropriate animation type to base your door off of, so you don't need to make the animation yourself. Export the model.

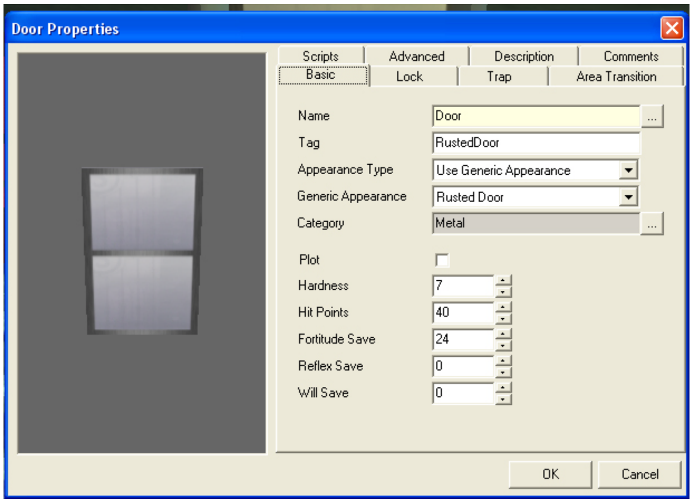


Import the model to 3DS Max. We are essentially going to retexture the door so it fits our purposes. I started by deleting the door knocker, because I didn't want it on my new door. You can add geometry to the door if you are familiar with hierarchies and relationships in 3DS Max, but I did not.

Create a new material for your door, and assign it. You may also need to edit the UV mapping, but I am not going to cover that in this tutorial.



Make sure all your models and textures are in the override folder or a hakpak, and load up your module. Your door is now in the toolset.

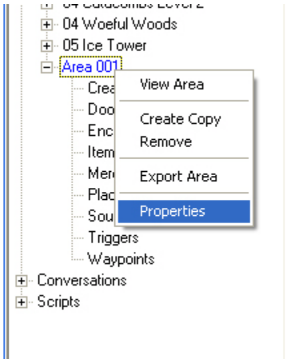


Name	Door
Tag	RustedDoor
Appearance Type	Use Generic Appearance
Generic Appearance	Rusted Door
Category	Metal
Plot	□
Hardness	7
Hit Points	40
Fortitude Save	24
Reflex Save	0
Will Save	0

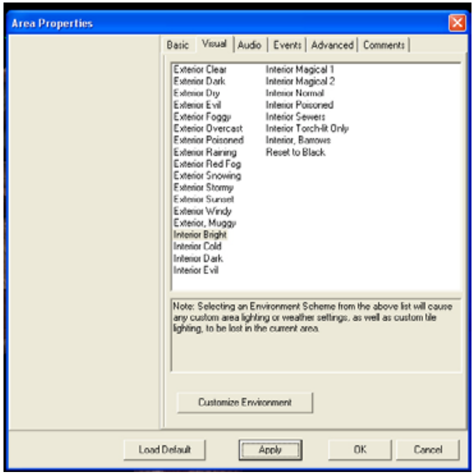
Custom Lighting Effects

The *Neverwinter Nights* Toolset makes custom area lighting simple, but the effect has such an effect on the visual appearance of the game it is worth pointing out.

Right click on your area, and go to Properties.

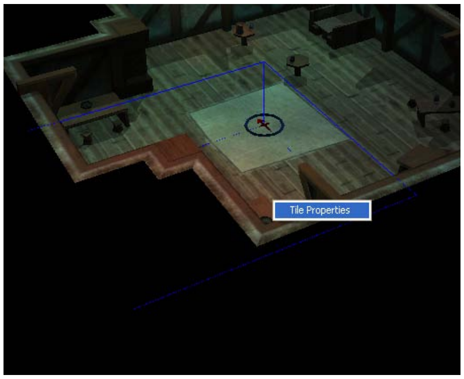


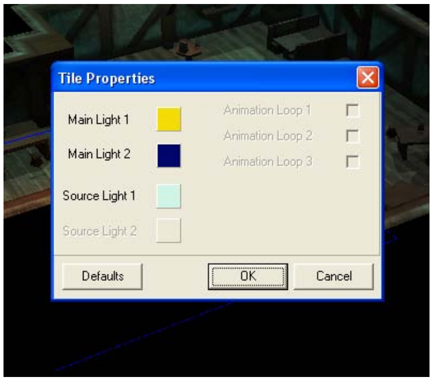
Under the Visual tab, there are many preset environments to choose from. Click apply to see what your area now looks like. Click on the Customize Environment button to get more options to adjust.



Note: Selecting an Environment Scheme from the above list will cause any custom area lighting or weather settings, as well as custom tile lighting, to be lost in the current area.

Once this is taken care of, you can right click on an individual tile and go into the tile properties to adjust the lighting for a particular tile. This generally isn't necessary, but sometimes you will want the extra control.



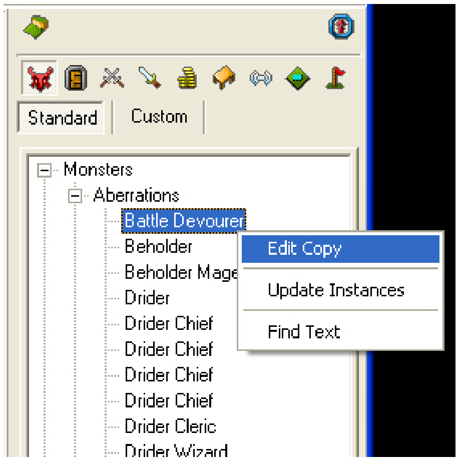


Creature Properties and Balancing Monsters

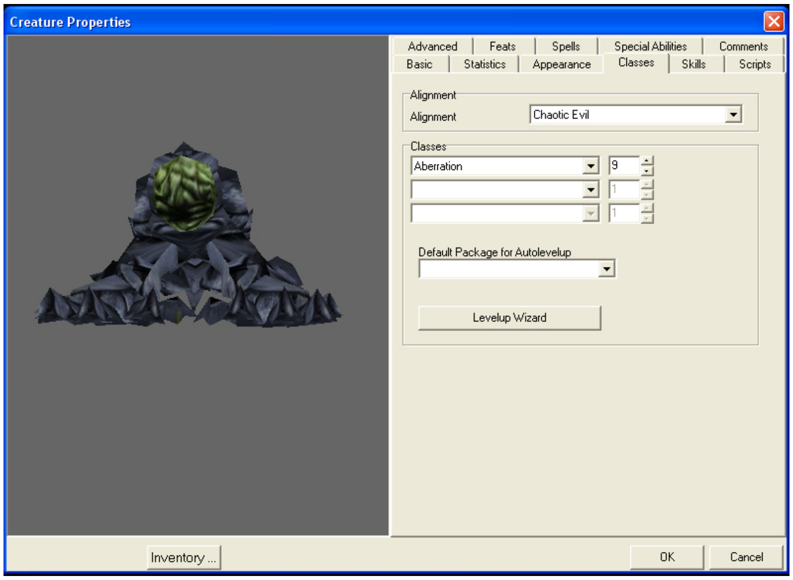
A normal *Neverwinter Nights* module includes a party of different classes of characters. This gives the player more options in combat, and allows them to play through the game however they want. However, our module leaves the player alone at various points of the game, so special care had to be made in creating monsters that provided an appropriate challenge.

In this tutorial, I will point out all the different things to consider when you are trying to determine how difficult a monster will be, and how to adjust them. However, you will have to playtest your module in order to know exactly how much a monster should be adjusted.

I will be using the Battle Devourer as an example, and point out all the important things to adjust. Right click on the monster, and choose Edit Copy.



This brings up the creature properties. Here we can see that it is currently at a Challenge Rating of 11. This is generally a good indication of how strong a monster is, but some attributes affect it more than others. Start by checking its level under the Classes tab.



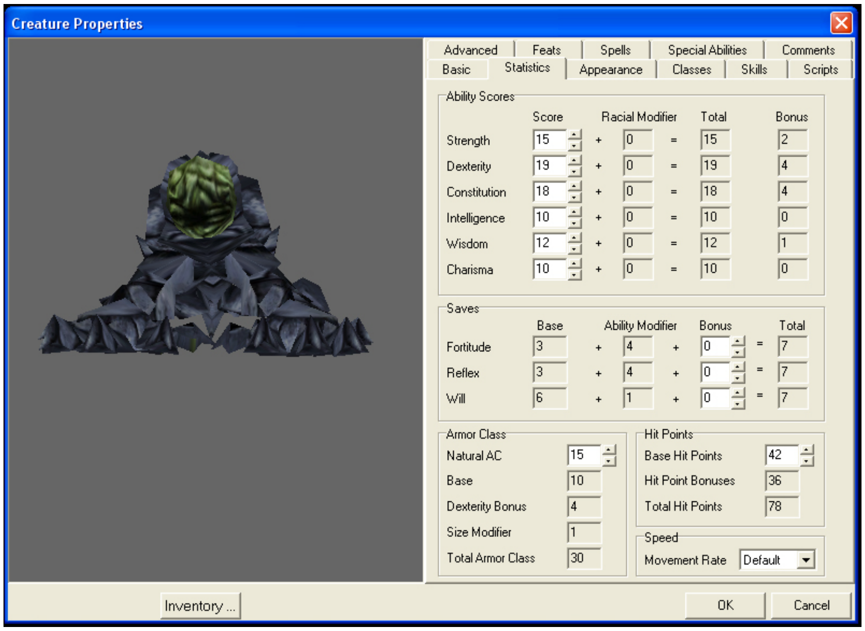
Alignment

Alignment: Chaotic Evil

Classes

Aberration 9

Creature level is a good place to start the adjustment process. Next let's check its stats under the Statistics tab.



Ability Scores

	Score		Racial Modifier		Total	Bonus
Strength	15	+	0	=	15	2
Dexterity	19	+	0	=	19	4
Constitution	18	+	0	=	18	4
Intelligence	10	+	0	=	10	0
Wisdom	12	+	0	=	12	1
Charisma	10	+	0	=	10	0

Saves

	Base		Ability Modifier		Bonus		Total
Fortitude	3	+	4	+	0	=	7
Reflex	3	+	4	+	0	=	7
Will	6	+	1	+	0	=	7

Armor Class

Natural AC	15
Base	10
Dexterity Bonus	4
Size Modifier	1
Total Armor Class	30

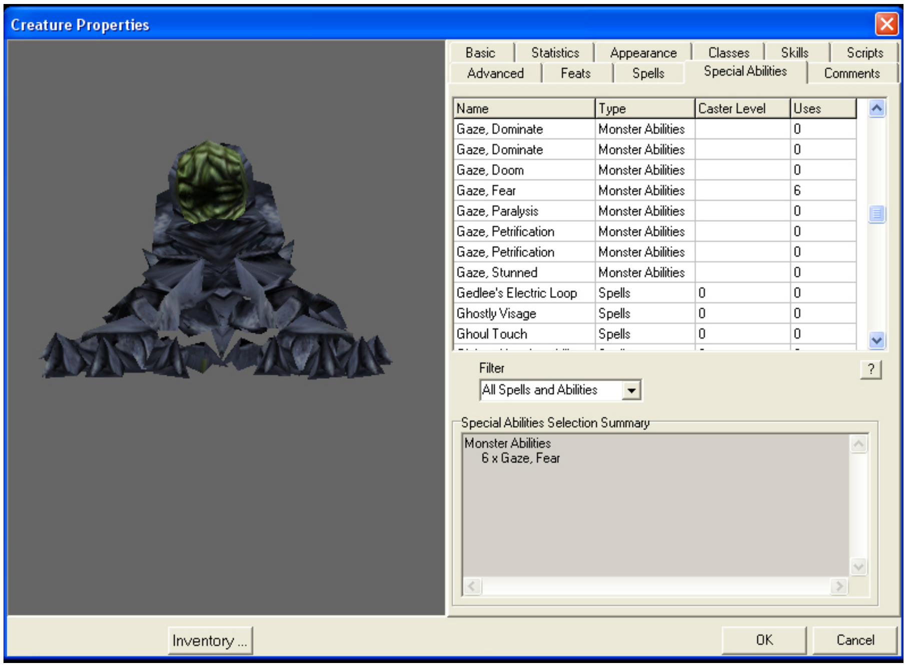
Hit Points

Base Hit Points	42
Hit Point Bonuses	36
Total Hit Points	78

Speed

Movement Rate: Default

Here we can see its ability scores, as well as its HP and AC. It is important to realize that these stats do not self adjust if you change a creature's level. So if you changed this level 9 creature into a level 1 creature, it would still have the Ability Scores, HP, and AC of a level 9 creature. Be sure to adjust this accordingly. Next is the Special Abilities tab.



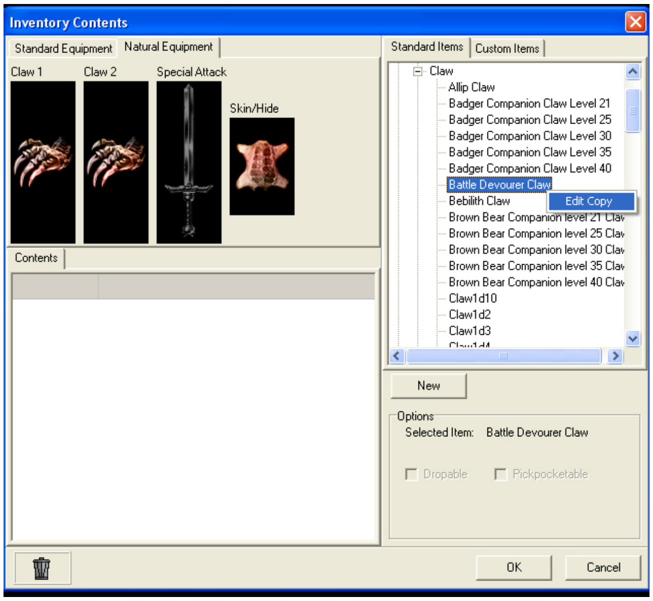
Name	Type	Caster Level	Uses
Gaze, Dominate	Monster Abilities		0
Gaze, Dominate	Monster Abilities		0
Gaze, Doom	Monster Abilities		0
Gaze, Fear	Monster Abilities		6
Gaze, Paralysis	Monster Abilities		0
Gaze, Petrification	Monster Abilities		0
Gaze, Petrification	Monster Abilities		0
Gaze, Stunned	Monster Abilities		0
Gedlee's Electric Loop	Spells	0	0
Ghostly Visage	Spells	0	0
Ghoul Touch	Spells	0	0

Filter: All Spells and Abilities

Special Abilities Selection Summary

Monster Abilities
6 x Gaze, Fear

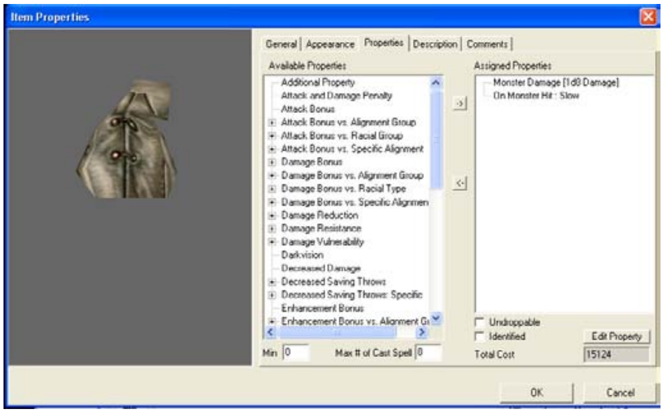
Some creatures have spells and special abilities. If it has a spell, make sure the caster level is appropriate for its level (if you have changed it). Also consider the effect the ability will have on the player in the module. In our module, where the player is often alone, fear is a devastating condition to be under. Things like this will become apparent during playtesting. Next, click on the inventory button under the creature model.



- Claw
  - Allip Claw
  - Badger Companion Claw Level 21
  - Badger Companion Claw Level 25
  - Badger Companion Claw Level 30
  - Badger Companion Claw Level 35
  - Badger Companion Claw Level 40
  - Battle Devourer Claw
  - Bebilith Claw
  - Brown Bear Companion level 21 Claw
  - Brown Bear Companion level 25 Claw
  - Brown Bear Companion level 30 Claw
  - Brown Bear Companion level 35 Claw
  - Brown Bear Companion level 40 Claw
  - Claw1d10
  - Claw1d2
  - Claw1d3
  - Claw1d4

Selected Item: Battle Devourer Claw

This creature doesn't have any standard weapons or armor, but it does have natural weapons. Right click on the blueprint to edit its properties.



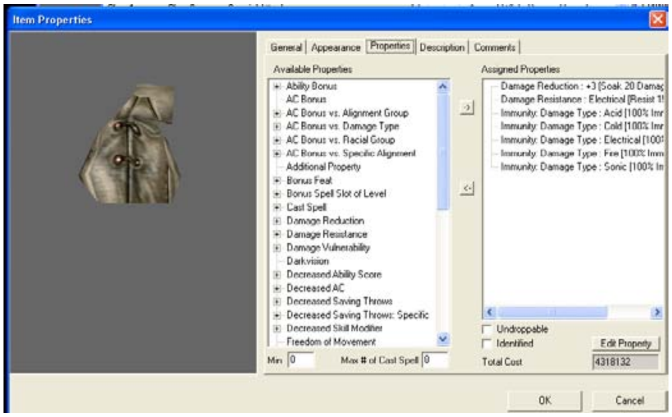
Available Properties:
- Additional Property
- Attack and Damage Penalty
- Attack Bonus
- Attack Bonus vs. Alignment Group
- Attack Bonus vs. Racial Group
- Attack Bonus vs. Specific Alignment
- Damage Bonus
- Damage Bonus vs. Alignment Group
- Damage Bonus vs. Racial Type
- Damage Bonus vs. Specific Alignmen
- Damage Reduction
- Damage Resistance
- Damage Vulnerability
- Darkvision
- Decreased Damage
- Decreased Saving Throws
- Decreased Saving Throws: Specific
- Enhancement Bonus
- Enhancement Bonus vs. Alignment Gr

Min 0   Max # of Cast Spell 0

Assigned Properties:
- Monster Damage [1d8 Damage]
- On Monster Hit : Slow

Total Cost 15124

Here we see what effect the creature's claws have, and change it if we wish. Next check the creature's hide properties.



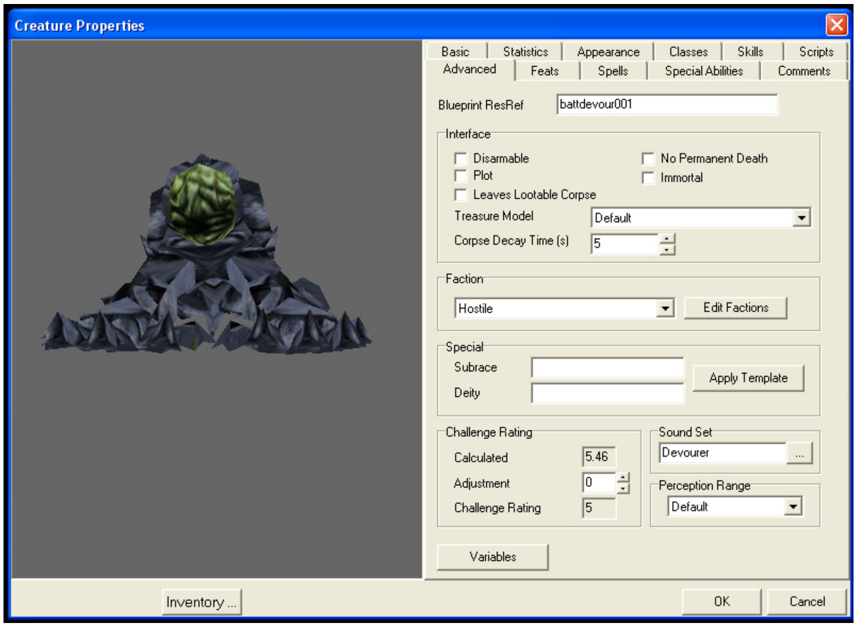
Available Properties:
- Ability Bonus
- AC Bonus
- AC Bonus vs. Alignment Group
- AC Bonus vs. Damage Type
- AC Bonus vs. Racial Group
- AC Bonus vs. Specific Alignment
- Additional Property
- Bonus Feat
- Bonus Spell Slot of Level
- Cast Spell
- Damage Reduction
- Damage Resistance
- Damage Vulnerability
- Darkvision
- Decreased Ability Score
- Decreased AC
- Decreased Saving Throws
- Decreased Saving Throws: Specific
- Decreased Skill Modifier
- Freedom of Movement

Min 0   Max # of Cast Spell 0

Assigned Properties:
- Damage Reduction : +3 (Soak 20 Damag
- Damage Resistance : Electrical [Resist 1!
- Immunity: Damage Type : Acid [100% Imr
- Immunity: Damage Type : Cold [100% Imr
- Immunity: Damage Type : Electrical [100!
- Immunity: Damage Type : Fire [100% Imm
- Immunity: Damage Type : Sonic [100% In

Total Cost 4318132

This creature is quite resilient to just about any kind of attack. Consider what level it should be at, and adjust accordingly. Finally, exit the inventory options, and go to the Advanced tab.



Blueprint ResRef: battdevour001

Interface
- □ Disarmable
- □ Plot
- □ Leaves Lootable Corpse
- □ No Permanent Death
- □ Immortal

Treasure Model: Default

Corpse Decay Time (s): 5

Faction: Hostile

Special
Subrace:
Deity:

Challenge Rating
Calculated: 5.46
Adjustment: 0
Challenge Rating: 5

Sound Set: Devourer

Perception Range: Default

Here you can manually adjust its Challenge Rating to affect how much XP it will give when killed. I inflated the Challenge Ratings of all the monsters in our module, so the payer would level up faster.

All that’s left is to playtest thoroughly and see how you fare against the creatures you have customized.

Appendex D: The Module

Works Cited

Alighieri, Dante, and Michael Mazur. The Inferno of Dante : A New Verse Translation, Bilingual Edition. Trans. Robert Pinsky. New York: Farrar, Straus & Giroux, 1994.

Bell, Aaron, Joshua Brunelle, and Garon Clements. "Veniss Underground: A Neverwinter Nights Modification." Advisor Dean O'Donnell. Worcester Polytechnic Institute (2008).

Chandigar. UDP2 Modern Office Interiors Tileset 1.3. HAK pack. UDP2 Modern Office Interiors Tileset 1.3 -- Neverwinter Nights 2 Vault. 1 Jan 2009. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Hakpaks.Detail&id=7596>.

DLA Team. DLA Canopied Forest. HAK pack. Neverwinter Nights 2 Vault. 31 Dec. 2003. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Hakpaks.Detail&id=3992#Files>.

Faloth the Unholy. 'Ultimate Creature Pak' v1.30. HAK pack. 'Ultimate Creature Pak' v1.30 -- Neverwinter Nights 2 Vault. 20 Aug. 2003. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Hakpaks.Detail&id=8>.

GuS. EtherPlanes - Rural Fire v1.0. HAK pack. Neverwinter Nights 2 Vault. 10 Apr. 2005. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Hakpaks.Detail&id=5532#Files>.

Helvene. Wooden Elven Interiors. HAK pack. Wooden Elven Interiors --Neverwinter Nights 2 Vault. 14 Jan. 2007. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Hakpaks.Detail&id=6662>.

Lisa. Lisa's Clothes Horse HotU Padded. HAK pack. Neverwinter News, Trailers, Screenshots, Previews, Reviews, Guides -- Neverwinter Nights 2 Vault. 20 Jan. 2007. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=hakpaks.detail&id=1742>.

Lisa. Lisa's Farm Girl Dresses. HAK pack. Neverwinter News, Trailers, Screenshots, Previews, Reviews, Guides -- Neverwinter Nights 2 Vault. 23 May 2005. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=hakpaks.detail&id=5812>.

Maximus. Neverwinter   News, Trailers, Screenshots, Previews, Reviews, Guides -- Neverwinter Nights 2 Vault. 12 May 2005. 27 Jan. 2009 <http://nwvault.ign.com/dm/scripting/course/Intro/RockAura01.html>.

Mayne, Robin. "Neverwinter Nights: Texture Tutorial: Re-Skinning Creatures." Neverwinter Nights: Welcome to BioWare's Neverwinter Nights (NWN) Community Site! BioWare. 27 Jan. 2009 <http://nwn.bioware.com/builders/texturetutorial.html>.

"Modifying Tileset - NWN1 Custom Content Guide." Main Page - NWN1 Custom Content Guide. 27 Dec. 2008. 27 Jan. 2009 <http://ccg.dladventures.com/index.php/Modifying_Tileset>.

Mr. X. Neverwinter   News, Trailers, Screenshots, Previews, Reviews, Guides -- Neverwinter Nights 2 Vault. IGN. 27 Jan. 2009 <http://nwvault.ign.com/?dir=resources/tutorials/MrX/&content=door_tutorial_new>.

Nauran, Pasus. NWN Area Load Screen Creation Kit v1.05. Tutorial. NWN Area Load Screen Creation Kit v1.05--Neverwinter Nights 2 Vault. 29 June 2005. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=other.Detail&id=266>.

"Neverwinter Nights: Console Commands." Neverwinter Nights: Welcome to BioWare's Neverwinter Nights (NWN) Community Site! BioWare. 27 Jan. 2009 <http://nwn.bioware.com/dms/commands.html#dm_givelevel>.

"Neverwinter Nights: For Builders." Neverwinter Nights: Welcome to BioWare's Neverwinter Nights (NWN) Community Site! BioWare. 27 Jan. 2009 <http://nwn.bioware.com/builders>.

"Neverwinter Nights: Neverwinter Nights Content Installer Creator." Neverwinter Nights: Welcome to BioWare's Neverwinter Nights (NWN) Community Site! BioWare. 27 Jan. 2009 <http://nwn.bioware.com/builders/content_installer.html>.

"Neverwinter Nights: Neverwinter Tileset Construction Tutorial." Neverwinter Nights: Welcome to BioWare's Neverwinter Nights (NWN) Community Site! BioWare. 27 Jan. 2009 <http://nwn.bioware.com/downloads/max_tileset.html#eula>.

Ob3ron. Ravenloft Shard Tileset Hakpack – Gothic City Tileset. HAK pack. Ravenloft Shard Tileset Hakpack – Gothic City Tileset --Neverwinter Nights 2 Vault. 12 Dec. 2007. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Hakpaks.Detail&id=60571&id=7239>.

Quantiko. Modern Cloth. HAK pack. Modern Cloth--Neverwinter Nights 2 Vault. 25 May 2007. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Hakpaks.Detail&id=7220>.

Sacateca, Eligio. "Custom Content Guide v3.0 -- Neverwinter Nights 2 Vault." Neverwinter News, Trailers, Screenshots, Previews, Reviews, Guides -- Neverwinter Nights 2 Vault. 01 July 2005. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Other.Detail&id=99>.

Schwartz, Alex. "Neverwinter Nights Custom Content Video Tutorials." GTProductions - Game Development Portfolio - Alex Schwartz. 27 Jan. 2009 <http://gtproductions.net/index.php?tab=comm>.

Scooterpb. MDL Plug-In Suite 1.0. 3DS Max Plugin. MDL Plug-In Suite 1.0--Neverwinter Nights 2 Vault. 29 March 2008. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Other.Detail&id=217>.

Smoo. "Neverwinter Nights: Buildtil.exe Issues." Neverwinter Nights: Welcome to BioWare's Neverwinter Nights (NWN) Community Site! 15 Jan. 2002. BioWare. 27 Jan. 2009 <http://nwn.bioware.com/forums/viewtopic.html?topic=189537&forum=48>.

Tiggs. Fancy Castle. HAK pack. Neverwinter Nights 2 Vault. 29 May 2008. IGN. 27 Jan. 2009 <http://nwvault.ign.com/View.php?view=Hakpaks.Detail&id=7509#Files>.

Unknown. Egyptian Crypt. HAK pack. Egyptian Crypt -- Neverwinter Nights 2 Vault. 27 Nov. 2002. IGN. 27 Jan. 2009 < http://nwvault.ign.com/View.php?view=Hakpaks.Detail&id=999>.

Additional Works Cited Information From the Artist

As the artist for this project, it is clear to me how much effort a person must put in to the creation of custom *Neverwinter Nights* content. I have created this section to more specifically point out the members of the online *Neverwinter Nights* Community whose work I incorporated into this project.

Chandigar, for creating the “UDP2 Modern Office Interiors” tileset. Although I retextured it entirely, it provided a basis and inspiration for the tileset created for our Eatery area.

GuS, for creating the “EtherPlanes - Rural Fire” tileset. I had to modify it quite a bit, but provided the base for our Scalded Plains area.

Helvene, for creating the “Wooden Elven Interiors” tileset. I completely retextured it, but it provided the basis and inspiration for our Pizzeria area.

Lisa and Quantiko for creating an assortment of modern day clothing items to use with our NPCs during the flashback scenes.

Ob3ron, for creating the “Gothic City” tileset used for our city of Dis.

Pasus Nauran for writing a tutorial on how to incorporate custom loading screens into a module.

The unknown creator of the “Egyptian Tombs” tileset. After a few modifications and some custom lighting effects, this tileset provided a basis for our final area, the Ice Tower.

All of the people whose work is included in the “Ultimate” series of hakpaks compiled by Faloth the Unholy. The .2DA files included in the “Ultimate Creature Pak” specifically was a big help when I was trying to figure out how to get custom creatures into the module. I even took a few creatures from this hak and incorporated them into our module.