

ANIMATED WEAPONS FOR NWN2

1.	How it works.....	2
1.1.	Weapons.....	2
1.1.1.	Standard weapons.....	2
1.1.2.	Baseitems.2da	2
1.2.	Animations	3
1.2.1.	For creatures	3
1.2.2.	For weapons	4
2.	Fix the Balor whip model.....	6
2.1.	Models.....	6
2.2.	Animations	6
3.	Create your own weapon.....	8
3.1.	Model	8
3.2.	Animations	9

1. How it works

1.

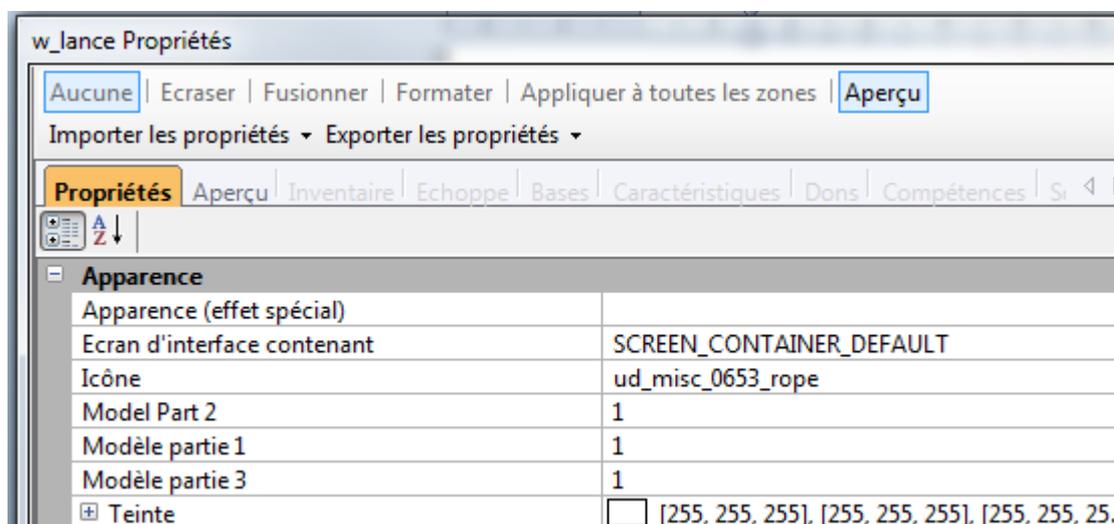
1.1. Weapons

There are already several tutorials explaining how to create weapons for NWN2, so I won't go into details. However, here is the basic knowledge to understand how to add animated weapons in game.

1.1.1. Standard weapons

Two kinds of weapons are available: customizable ones, for which you can select how the handle, hilt and blade will look like (obviously this relates to a sword). For NWN2, these parts are labelled Part 1, Part 2 and Part 3 and for custom content, creators will have to respect the following naming convention:

- If the weapon type is named "w_weapon", the various models shall be named "w_weapon01", "w_weapon23", etc...
- Each part will bear the same name appended with **_a** for model part 1, **_b** for model part 2 and **_c** for model part 3: in the case of "w_foil45", we'll have *w_foil45_a* for the blade, *w_foil45_b* for the hilt and *w_foil45_c* for the grip.
- These parts can be selected independently from each other, so it's possible to create a sword with **w_sword01_a** blade, **w_sword02_b** hilt and **w_sword03_c** grip. This means any new model must at least have a size similar to the family of the weapon it will complete.



The alternative is to create a weapon using a single mesh (in other words, handle, hilt and blade will always be together, in a whole model). In this case, the naming convention is "w_weapon67" (no letter appended).

The way NWN2 distinguished three-part-models from whole model comes from baseitems.2da.

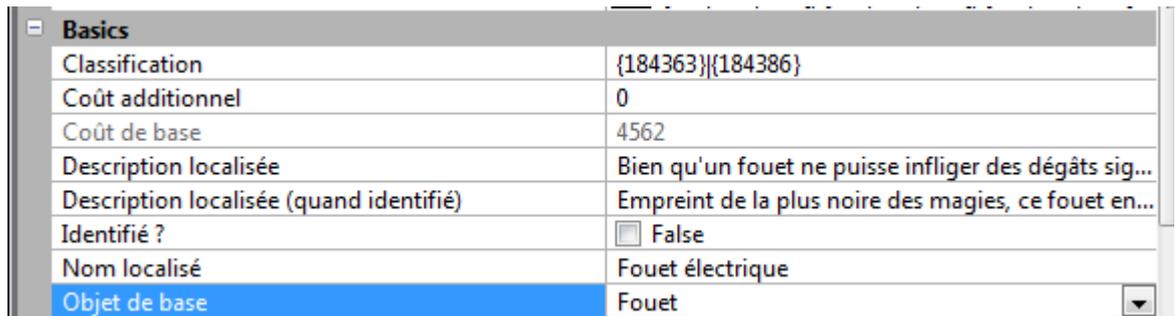
1.1.2. Baseitems.2da

The two dimensional array (in plain English, a table with several lines and several columns: each line concerns a given item, and each column defines a property of the item) *baseitems.2da* lists all the items characteristics used by NWN2.

Name	label	InvSlotWidth	InvSlotHeight	EquipableSlots	CanRotateIcon	ModelType	NWN2_Anim	ItemClass	GenderSpecific	Part1EnvMap	Part2EnvMap	Part3EnvMap	DefaultModel	NWN2_DefaultIcon	DefaultIcon	Container	WeaponWield	WeaponType	WeaponSize
4	176 lightflail	1	3	0x1c030	1	2	1	1_w_flail	0	1	1	1	1_it_bag	9	****	0	****	2	3
31	1525 dart	1	1	0x00010	0	0	0	0_w_dart	0	1	1	1	1_it_bag	33	****	0	11	1	1
35	**** DELETED	2	4	0x1c030	1	2	1	1_w_flail	0	1	1	1	1_it_bag	37	****	0	****	2	4
61	1555 sling	1	2	0x00030	0	0	0	1_w_sling	0	****	****	****	1_it_bag	57	****	0	10	2	2
111	83617 Whip	2	2	0x1c030	1	0	1	1_w_devilwhip	0	1	1	1	1_it_bag	64	****	0	0	3	2
201	Fouet Fouet	2	2	0x1c030	1	0	0	1_w_fouet	0	1	1	1	1_it_fouet	64	****	0	****	3	2

What we're more interested in here is:

- Label → what will show in the “base object” dropdown list:



- ModelType → 0 for weapons designed in a single mesh, 2 for weapons using three different parts.
- NWN2_Anim → 0 for rigid objects, 1 for items with animations.
- ItemClass → name of the model (the .mdb file)
- GenderSpecific if the appearance of the object changes with the gender of its wielder (0 = no, 1 = yes).

This table is where all data regarding the objects any creature can wield is stored. See line #111 for the whip? This is also where we will change the ItemClass (see next pages).

1.2. Animations

In NWN2, animations need a skeleton (defining how the “bones” are linked together) and animation files (defining how the bones are oriented and move with regards to the origin bone).

This is true for creatures and items (not placeables!) alike.

The only difference can be found in the naming conventions.

1.2.1. For creatures

The body for the creature named “*c_monster*” has to be named “*c_monster_CL_Body01*” (where 01 is a number indicating the variant, this can be selected in the “armor appearance” tab in the creature’s properties).

Its skeleton has to end with *_skel* (for the sake of clarity, it is usually named after the creature, thus “*c_monster_skel*”, but this is not actually compulsory).

What is compulsory however is that the animation files refer to the skeleton (if the skeleton for “*c_monster*” is “*c_creature_skel*” for instance, all the animation files will have to start with “*c_creature_*”).

They also need to clarify which animation they’re used for:

- Idle for the “doing nothing” animation,
- 1attack01 for the attack animation (first variant, the second will be 1attack02 and the third 1attack03 etc...).

Finally, they have to specify which combat stance is concerned (“1hS” for one-handed weapon, “1hSS” for one-handed weapons with shield, “D2h” for two-handed weapons etc...)

Let’s take an example: our creature “*c_monster*”, whose body is “*c_monster_cl_body03*” which uses the skeleton “*c_creature_skel*”, will use the animation “*c_creature_1hSS_1attack04*” for the fourth variant of the one-handed weapon with a shield attack.

1.2.2. For weapons

Notes:

- These are my findings and might not be applicable in every case.
- Weapons being defined in the same 2da as the other items, it’s possible that blowing flags (as items, and not creatures) or books with pages flipping be added with the same principles.
- A model named as per the creature scheme (“*c_creature_CL_Body01*”) and the same model but this time named as per the weapon pattern (“*w_weapon01*”) are necessary: the “creature” for the animations and the “weapon” for the appearance (at least in the toolset).

The requirement for a skeleton remains. The naming conventions for the animations files are modified as follows however (for ease of reading, it is assumed that “*w_weapon*” is the name used for the weapon for which animations are designed):

- In addition to the “*w_weapon*” prefix, the animations files will need to include in their name a letter F or M (for female/male wielder, see [1.1.2](#) about gender specific appearance),
- After the letter F/M, a second letter L/R (for left/right use, R is probably the default animation and L only used in case of dual wielding). Even for two-handed weapons, the FR or MR part is compulsory,
- Then the combat stance as for the creatures (*_UNA_*, *_1hs_*, etc...),
- And finally the type of animation (same rules as for the creatures: “idle”, “1attack01”, “death02”, etc...).

Additionally: the above assumes that the standard animations apply. If wielding the new weapon requires a specific animation for the attacker too, it obviously has to be created and named to reflect the type of attack and the type of weapon. For the sling for instance, it gives *P_??M/F_Thrw_slingattack01* where ?? is one of the four possibilities for players [DD for dwarves ; GG for gnomes ; OO for half-orcs ; HH for humans, elves, halflings – basically all other playable races].

I haven’t checked, but that probably means that trolls won’t be able to use slings (or rather, their attack animation will be weird).

A female drow attacking with a whip (whose model is named “elfwhip”) and using a shield will need a *P_HHF_1hSS_elfwhipattack01* animation if the usual sword-like moves were to be avoided. But this is purely guesswork, I haven’t checked this either.

The question that remains unanswered is: why does the Balor whip still not work after all the above has been done?

2. Fix the Balor whip model

The answer is that there were many requirements not met in the stock version:

- The name of the mesh inside the .mdb was not the same as the name of the .mdb;
- The scale of the whip was not the same as the scale used for the skeleton;
- The animations were named for a creature, not for a weapon (the FR / FL / MR / ML part was not present).
- The name of the base object in baseitems.2da (w_whip) was not consistent with the names of the animations (w_devilwhip_XXX).

2.

2.

2.1. Models

First thing to do: use the same name for all the stuff used by the whip (mesh, model, skeleton, animations). As we don't want to redo all the animation, let's take the skeleton name as the reference, and change baseitems.2da line #111 to read "w_devilwhip" instead of "w_whip".

Since the 2da now refers to a **devilwhip.mdb** model, let's make sure there's one (w_whip, w_whip01 and even w_whipdevil exist, but they unfortunately all refer to w_whip or w_whip_cl_body01 whereas baseitems.2da specifically told the NWN2 engine to look for **devilwhip.mdb**, **devilwhip01.mdb** [ModelType = 0, so only one part expected] and **devilwhip_cl_body01.mdb**).

Here's the procedure:

- ✓ Rename w_whip01.mdb to w_devilwhip01.mdb.
- ✓ Open w_devilwhip01.mdb.
- ✓ Rename the mesh w_whip to w_devilwhip01.
- ✓ Export at 5500% of its original size (including the hook points hp_0, hp_1 and hp_2).

- ✓ Open w_devilwhip.mdb.
- ✓ Rename the mesh from w_whip to w_devilwhip.
- ✓ Export at the original size (no need to include the collision spheres).

- ✓ Open w_DevilWhip_CL_Body01.mdb.
- ✓ Rename the mesh from w_whip to w_devilwhip_CL_Body01.
- ✓ Export at the original size (include the collision spheres this time).

Now, the Balor whip files are consistent: the animated model, the weapon model, the skeleton and the animations all use the same name "w_devilwhip".

2.2. Animations

As far as the character is concerned, the animation should be _1hS_ or _D2h_ etc... However, as far as the weapon is concerned, we're considering an UNArmed stance. Consequently, renaming all w_DevilWhip_D2h_animation to w_DevilWhipFR_UNA_animation will allow using them in game.

Note: I still haven't figured out why all the Balor whip animations are supposed to be UNA. After all, the flail and sling animations follow the rules described in [1.2.2](#).

I noticed however that the animations for the Balor whip included references to both "w_devilwhip_skel" (which is expected!) and "w_whip01" (which is quite surprising...) I'd bet it's the reason.

Anyway, if you export your own animations without the mesh, there shouldn't be any problems.

3. Create your own weapon

For details on how to create your own models or animations, there are several tutorials available – one I like particularly is by Frank Perez (aka Elysium), which can be found on his blog <http://nwn2faithless.blogspot.com> (search for “creating new life with Expotron and max”).

3.

3.

3.1. Model

As you have probably guessed, the way animations work for weapons means that only model number 01 will be shown in game. In case you don't believe me, try to equip a flail while in the toolset:



And see what it looks like in game:



Visual effects:

At this time, hp_0 and hp_1 don't give the expected results... This apparently is a bug within the engine, as the flail (the other animated weapon with the sling in the stock game) also shows the visual effect all over the weapon, and not at the spiked ball only (as it's supposed to be - just compare with the torch to get the idea).

The good news is that "creature attachment points" do work: add an "ap_forehead" or "ap_mouth" or whatever hook point you'd like to the skeleton, and create the visual effect with the same attachment points as reference (the creation of visual effects is not in the scope of this tutorial, but you should find quite a few good tutorials easily).

3.2. Animations

As explained in [section 1.2](#), there's a major difference between the naming convention for creatures and weapons: creatures' animations refer to the skeleton they use, whereas weapons animations also refer to the name of the model.

Let's take an example:

Wraith spiders have a model named *C_SpidWrai_CL_Body01*, but like all spiders use the *c_spid_skel* skeleton and therefore the *c_spid_una_idle*, *c_spid_una_1attack01*, etc... animations.

The Balor model on the other hand, using *w_devilwhip_skel*, didn't work initially as its mesh was named *w_whip*: there was an inconsistency between the skeleton, the mesh and the model names.

To summarize: only one model can be used, and the same name is used for:

- The skinned model: **w_weapon_cl_body01** (used for the animations)
- The rigid model **w_weapon01** (used in the toolset for the appearance of the weapon)
- The skeleton: **w_weapon_skel**
- The animations files: starting with "**w_weapon**" (don't forget the gender and side before the stance and animation, as explained in paragraph 1.2.2).

- Visual effects need “ap_xxx” hook points, and not the usual hp_0, hp_1, etc...