

H&D2 model editing tutorial (excerpt)

This is a translated version of the German tutorial by Ted Striker.

It includes a part of the additional content for advanced modders of the 2011 version.

For modeling basics please see the tutorials by Zdenda and Hawk or the complete model editing tutorial (only in German).

To be continued / supplemented

Creating new models

If you want to create new models, e.g. a supply box with altered textures, copy the model file in question, e.g. la_ammo01.4ds from the extracted models folder, rename it and put them in the models folder in the main program directory of Hidden & Dangerous2

The new file name should have the same number of characters as the original file, so the models can be easily replaced in an actors.bin. Then you use hex editor to put in the file names of the changed textures in the new 4DS file. Also the file name of the new textures should have the same number of characters as the original files.

Changing models with a hex editor

If a file or a 4Ds model scene.4ds is opened with a hex editor, you are at first confronted with a seemingly meaningless jumble of letters and numbers.

The following instructions may sometimes sound a bit complicated, but if you have understood, how to use the editor, you can do really cool things with it;)

The Hexadecimal editor

Hexadecimal refers to the priority number system based on 16, using 16 instead of the ubiquitous 10 digits. The "digits" 10 to 15 are represented as the letters A, B, C, D, E, F.

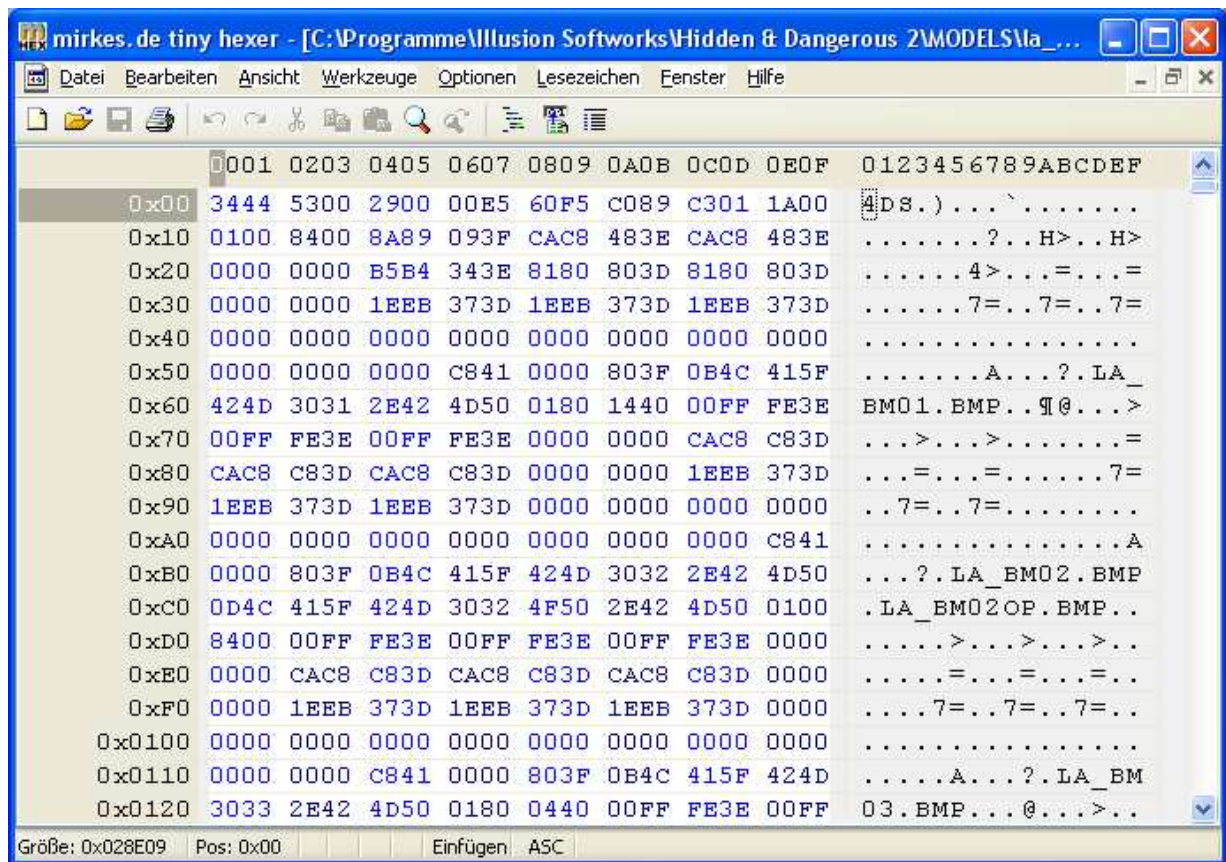
Hex	Decimal
A	10
F	15
10	16 (1x16)
2A	42 (2x16 + 10)
100	256 (1x16 ²)

If you have a 4DS file opened in hex editor you see the following three-part structure of the editor.

On the far left for each row, the display position (offset), starting with 0x00.

Next a field of 16 by 2 points, upon which stands 00 to 0F (in this case, the bytes in hexadecimal)

On the far right a field of 16 points, above them 0 to F. Here, the bytes are represented as text in a selectable character set. To improve clarity in the right pane, you can set the View / Character conversion to ASCII 7 bit.



In this field right you recognize entries in plain text, such as LA_BM01.BMP in the case file for the motorcycle la_BmwR75.4ds. The textures used are listed at the beginning of a 4DS file. To put in the filename of a new texture file it is best to use the Search / Replace function under Edit. If looking for a plain text, box "text" must be enabled.

If the box is "text" is not enabled, you can replace sequences of numbers (shown in the middle field), such as for equipment in an items.dat file, like this:

Search: 00000004 (K98)

Replace: 00000012 (MP40)

Essential that the character length is not changed, otherwise the file is no longer usable.

Creating new vehicle models

For vehicles available there are at least 3 other models with the same file name and other letters in front, e.g. for la_citroen.4ds

cla_citroen.4ds (wreck Version)

DMLa_citroen.4ds (damaged)

sla_citroen.4ds (shadow effects)

These models, which are found in the extracted models folder with the file search function must also be copied and renamed, if you want to create a new vehicle model, otherwise it will not work. Example for new file names:

la_citro01.4ds (new vehicle model changes only entered here)

cla_citro01.4ds

DMLa_citro01.4ds

sla_citro01.4ds

The components of a 4DS model

All models, which also apply to the landscape, which are scene.4ds in the missions folder, include the entries for the used textures in the beginning and are composed of several sub-models.

These components are subsequently referred to as sector and are not to be confused with the ~~above-mentioned~~ sectors for the light. For cars, there are usually the following names for sectors:

WHL00 left front wheel

WHR00 right front wheel

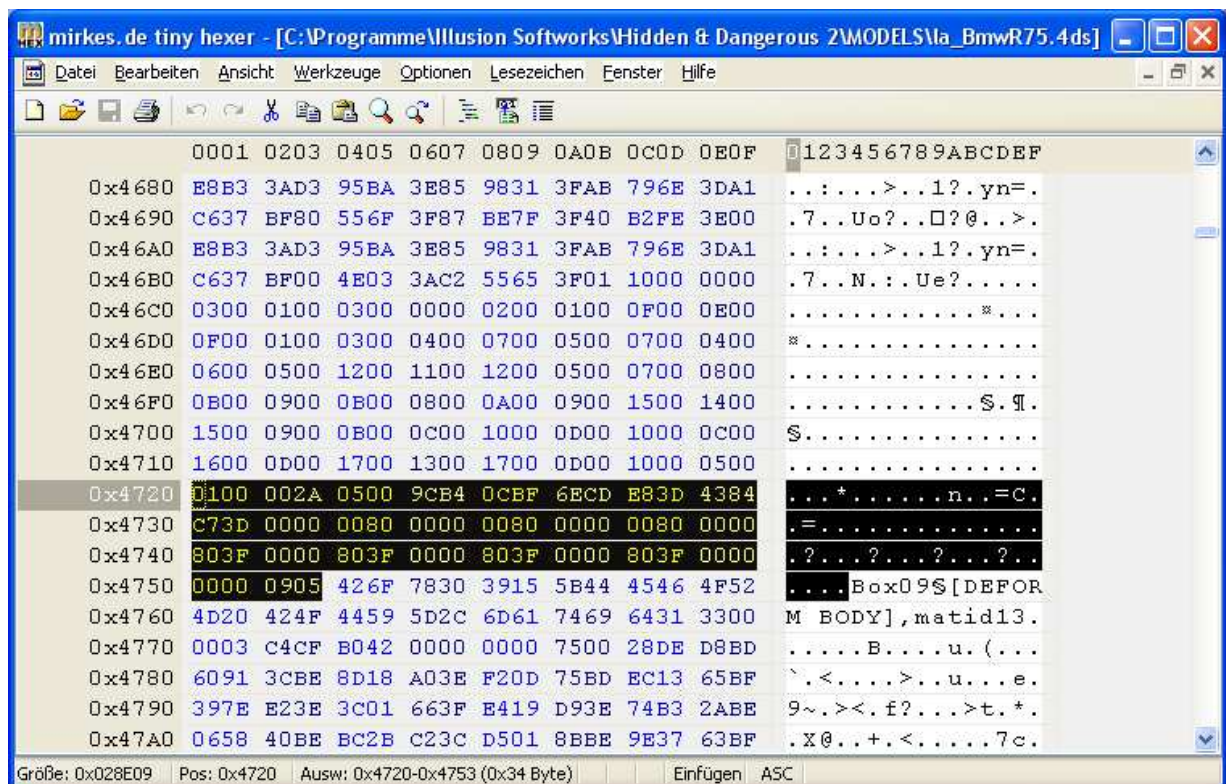
WHL01 rear left wheel

WHR01 rear right wheel

Telo (Czech body)

The other sectors have once Czech, English names sometimes, often simply names like Loft, Mesh, line, box, etc.

How to find now, the individual components, i.e. sectors of models without having to search in the confusing maze of characters in a file opened in hex editor? For this purpose one must know that before every sector is a kind of definition, which always begins with the sequence of numbers 0100002A. Therefore, we look after this series of numbers and get in the editor to the definition of the sectors. A definition consists of 3 lines and 8 points in the central display of the editor, before the name of each sector.



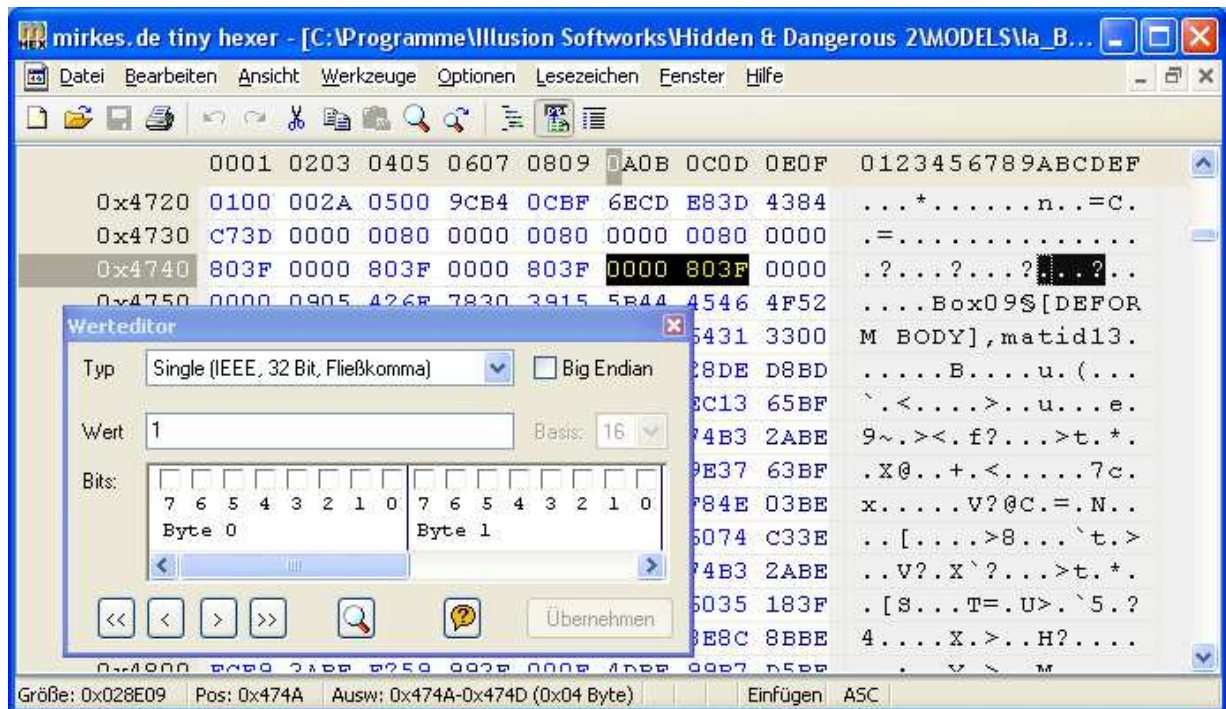
This includes some places that cannot be changed or where it is unclear what happens if you change it. This can go up to the uselessness of the file.

Included are also a total of 10 variable parameters / values for position, rotation and scaling of the sector included in the model. Each value is assigned 8 points and is coded as IEEE 754 floating-point number in little endian format.

In 4ds file	decimal
0000 803F	1
0000 0040	2
0000 00C0	-2
0000 003F	0,5
0000 8043	256

In this numerical system there is +0 (0000 0000) and -0 (0000 0080), and minor inaccuracies, it can be a value such as 0.1 represented only as approximation. These tolerances are not important for the game, because they occur only after the umpteenth decimal place.

If you want the values about to change remain clear and precise, you do not scale in tenths of steps, but with the reciprocal value of powers of 2, e.g. $1/8$ corresponds to 0.125



With the value editor of Tiny Hexer (second icon from right below the menu bar) you can enter values directly. Select type Single (IEEE 32-bit, floating point) and deselect the box next to Big Endian.

Now go to a value such as 0000 803F in the hex file and mark it. Now, in the "value" field the decimal value is displayed, here it is 1. If you now enter in the "Value" field a different decimal value, it is assumed to be converted, to do so click on "Apply changes". It is important that you always just mark the 8 digits of an editable value with the cursor and no other ones. Again and again, back up the file!

Editable values of a definition

For most standard models you can change position, rotation and scaling of individual sectors. That's not for files with special characters like # at the beginning, containing an animation. Even for objects that cannot be directly positioned as a bridge from Burma1, it may be different.

Here again is the def of a sector within the la_BmwR75 model. The sector is the engine and named Box09.

The Def begins with the characters 0100 002A (unchangeable)

After that four unalterable places 0500

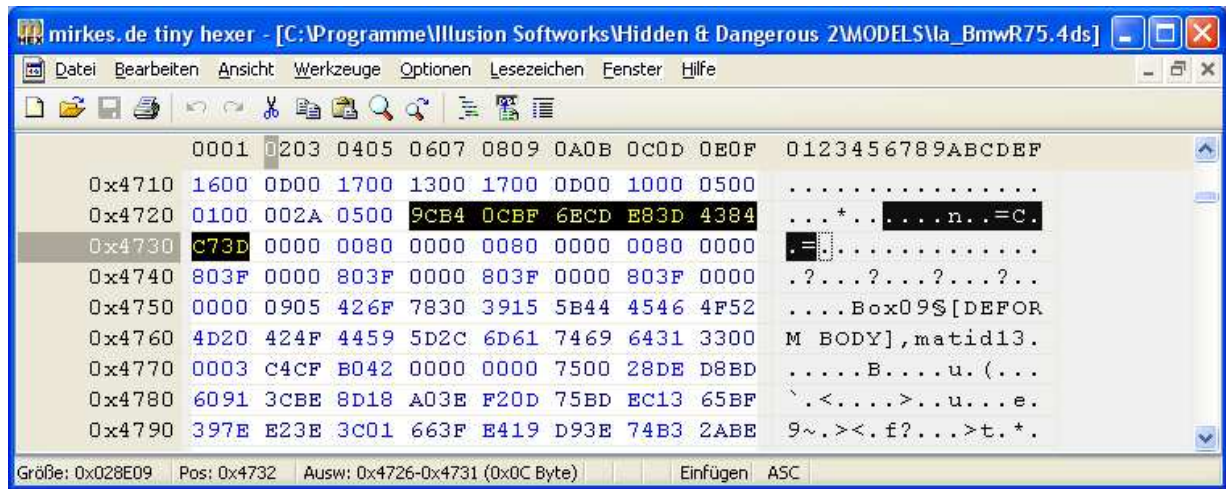


Next are three values for the position X, Y, Z

9CB4 0CBF (Pos. X about -0,55)

6ECD E83D (Pos. Y about 0,11)

4384 C73D (Pos. Z about 0,1)



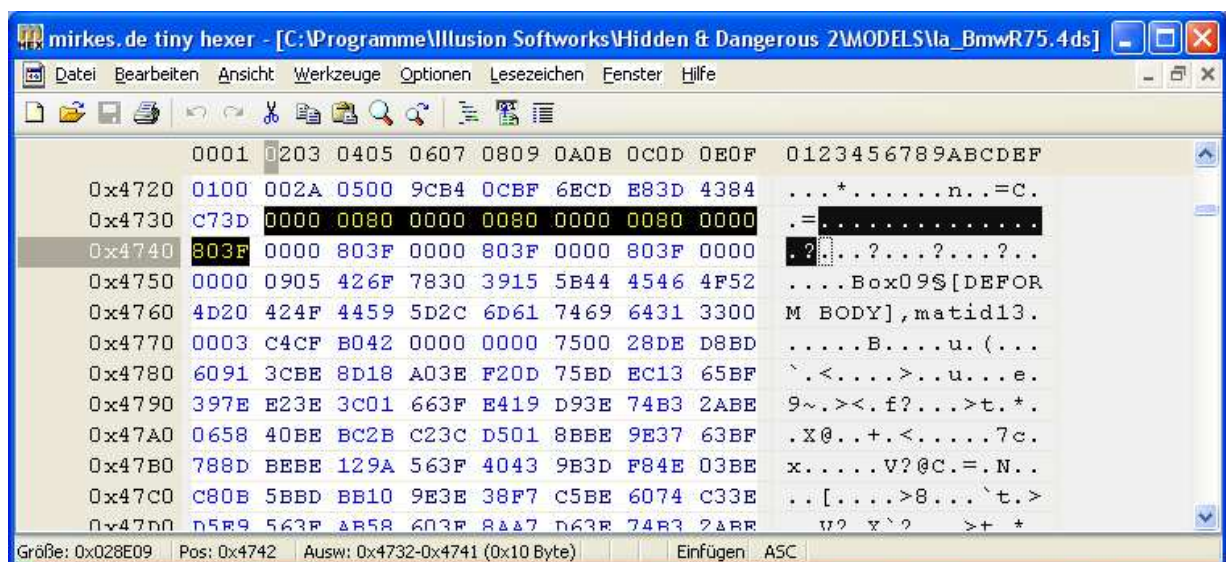
Then follow four values for the rotation (there is an additional axis for a global coordinate system)

0000 0080

0000 0080

0000 0080

0000 803F

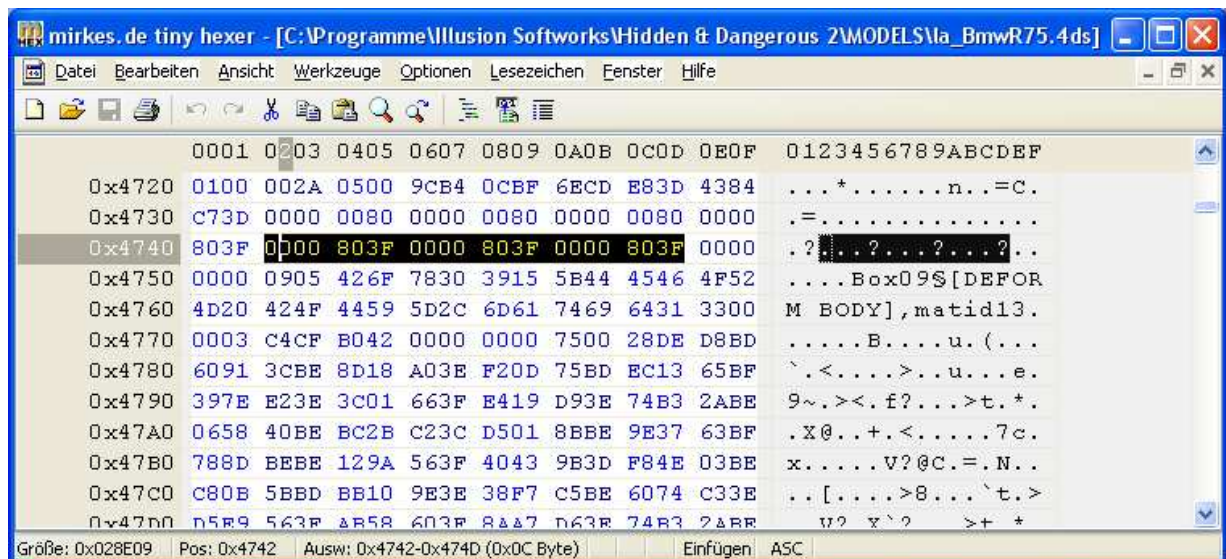


Then come the three values for the scaling

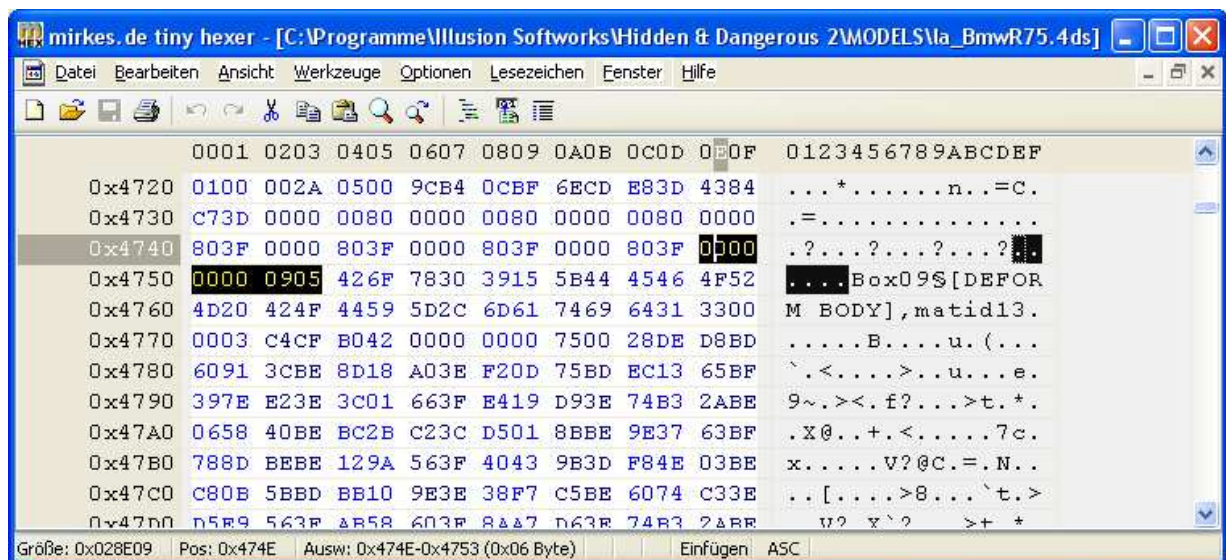
0000 803F (Factor X=1)

0000 803F (Factor Y=1)

0000 803F (Factor Z=1)



Finally, 3 by 4 points unchangeable digits 0000 0000 0905 before the name "Box09"



Coordinate axes for vehicles:

X transverse to the direction of travel

Y vertical axis

Z travel direction

Depending on the rotation, the values for X, Y, Z can also in a different order, but usually like this.

The values for the rotation, you cannot enter directly any angle from 0 to 360 degrees. How you can enter values here and how it behaves with the global axis is not yet clear. (Quaternions)

The scaling values are usually three times 0000 803F, which corresponds to the decimal value of 1. To find out where a sector of the models is, you can increase the Y value from 1 to 5, and then recognize in the game the distorted object, what it represents, for example, engine, wings or whatever.

Edit: with the Plug-in by hdmaster, you can now import 4DS models into a 3D program such as 3DS Max or GMAX and see the sectors and dummies of a model.

By changing of values new vehicles can be created on the basis of available vehicles from H & D2, e.g. (Enlarge Scale X value) with bigger wheels. Otherwise, you can also remove parts such as fenders, bodies, doors etc, when you set all three scale values of a sector to 0.

The collision of scaled to 0 sectors remains as before.

The front axis of a vehicle you should not move, because otherwise the wheels are shifting while driving to the outside and the body does not move properly.

Dummies like SEATs or for a barrel or turret of a mounted gun named HORT00 and VERT00 can also be changed (other position), but they do not start with 0100002A. The value positions are the same.

Texture of a sector

Each sector is assigned one of the textures that are listed at the beginning of the file. For large landscape sectors, this is usually only the main texture; a sector may also have other textures. The assigned texture is resulting from a value at the end of a sector. This is a hexadecimal number that is related to the order of the textures at the beginning of the 4DS file. It is encoded in the middle field of Hexeditor, the first 2 of 4 digits before the beginning of the next sector, which manifests itself in the number sequence 0100002A.



Attach parts to vehicle models

A vehicle can be extended with other models that are then firmly connected with him. Do this by creating an additional object with *add object* function, e.g. a box lying on the bed of a truck.

In DCED under sector put in the name of the vehicle model object and its sector, with what the new model is to be combined, separated by a dot. Case sensitive.

The additional models may not be too complex; otherwise the Def of the vehicle is broken. The additional model is better attached to a fixed sector like Telo and not to a sector, which may be separated due to damage, such as fenders, etc.

There are a few models, such as for example la_BmwNah01.4ds of individual vehicle components such as doors, headlights, fenders, etc. These files all have the letters "Nah" in their name and are easy to find.

Excerpt of HD2 Model Editing Tutorial by Ted Striker. No final edited version, use at your own risk.