Welcome to WinTex 4.2

INTRODUCTORY

About WinTex 4.2.
Installation of WinTex 4.2.
The WinTex Tutorials.
WinTex   Registration.

The main features of WinTex:

EXAMINE WAD FILES

WinTex will let you browse almost every entries in the WAD files.
- View **pictures** and **textures**,
- Listen to **sounds**, and musics
- View the level **maps**, the level **BSP tree nodes**, the level **Reject data**
- View miscellaneous **data** lumps.

MODIFY WAD FILES

WinTex will let you Edit and Test WAD file in convenient way:
- modify **sprites**, **graphics**, **sounds**, **musics**, **textures, floors, sky**.
- **cut** and **paste** entries between WADs
- **reorganise** your WADs.

MODIFY GAME LEVELS

WinTex also help you edit level maps:
- Modify the **textures, floors** in level maps
- Rebuild the **reject** data and make special effects (like blind monsters)
- Rebuild and **check** the BSP tree **nodes**.
- Modify the level **maps**, by using the level editor of your choice.

COMBINE WAD FILES

WinTex/DeuSF will let you Combine two or more existing WAD files into one.
- Properly merge modified **sprites**, **flats** and **textures** (no other tool does that).
- Play   **DOOM** wads under **Ultimate DOOM**.

MODIFY THE MAIN WAD FILE

WinTex/DeuSF let you Modify the game files of DOOM, DOOM2, HERETIC or HEXEN.
- Your favourite WADs will **permanently** be present in the game.
- You can get back your **original** (unmodified) game file at anytime.
- You can even get back the WAD you put inside.
-   The **only safe way** to modify the main game file (beware of other more primitive tools)

Registering WinTex

WinTex is user supported.

This program is not a real shareware, since you have no obligation to register.

By registering you indicate that you're interested in the program, and that you would like to contribute to it's improvement. You also greatly contribute to the motivation of the author.

Of course, you may believe that the hard work of other people should be available to you for free, and that I owe you support, bug fixes and improvements. Well, no problem. Just ask... and you will be flamed on sight.

Registration

Via Compuserve

GO **SWREG**
Select **Register Shareware**.
Enter program identification: **7683** (for WinTex 4.x).

Via mail

Send your registration fee of 10$ (U.S.) in a self-adressed letter to the following address:

Olivier Montanuy
9, rue du Moulin
Ploumanach
22700 Perros Guirec
France

About WinTex 4.2

What is WinTex?

WinTex 4.2 is an **editor** for the games files (.WAD) of **DOOM**, **DOOM2**, **Ultimate DOOM**, **HERETIC** and **HEXEN**, by id software and Raven software.

WinTex 4.2 is also a **browser** the games files (.GRP) of **Duke Nukem 3D** by 3D Realms.

WinTex 4.2 does not support Quake editing. That will be for later versions.

This tool is the result of the work of many people: please also take a look at the Thank list.

How can WinTex help you edit DOOM and HERETIC?

WinTex is a tool to create or **modify** the wall **textures,** the **pictures,** the **sprites,** the **sound effect,** the **musics,** the **floors** and **ceilings** that are used by **DOOM** and **HERETIC.**
This tool is not a level editor: there are lots of very good level editors already.

With **WinTex**, you can concentrate on your creations, and forget about most of the tricks and bugs of **DOOM** editing.

WinTex contains all the features needed to create WAD files with custom textures, and to generate and install WAD files with modified **sprites** and **floors/ceilings**.

Don't forget WinTex is shareware, register if you like it to become even better.

How comes WinTex makes editing easy?

WinTex is **an integration tool**. It is not a graphic editor or a sound editor by itself. With WinTex, you will use your favourite graphic editor or sound editor, because WinTex takes common BMP picture, WAV sound, or text files, and **converts** them to the special DOOM format.

WinTex also help you **organise** that data, so that DOOM can use it correctly.

WinTex includes features that are specific to DOOM, like Texture editing, Name identification, Reject map editing, Filtering of the different entry types.

WinTex handles Levels, Lumps, Sound, Textures, Graphics, Sprites, Patches, Flats, Music.

Moreover, all editing under WinTex is reversible. Just press 'restore WAD' and the WAD you had been modifying drops back to it's previous state. Of course, this waste data, so WinTex has a WAD optimiser to create WADs with no wasted data.

Last, WinTex will help you use some usefull but cryptic tools available on the net, like the Reject builders, BSP builders and end screen editors. There are many DOOM and HERETIC editing tool around, each specialised in it's own domain. WinTex is here to make them work together efficiently.

Data Utility License

Due to the very low stability of Windows 3.1 and Windows 95, this program comes with **no warranty whatsoever**. Use it at your own risk. **No liability accepted**.

DOOMtm and HERETICtm are trademarks of **id Software,** Inc.
The DOOM and HERETIC characters and the Id Software logo are trademarks of Id Software, Inc. DOOM Copyright (c) 1993. **id Software** Inc.
All rights reserved.
This utility product is **not a product of id Software, Inc.** and has not been approved by id Software, Inc.
id Software will not provide technical support for this product.
The purchaser of this utility product **may not distribute for money** or other consideration any software data created and/or modified by the use of this utility product.

Thanks list

WinTex is copyright (c) 1994, 1995 and 1996 Olivier Montanuy.
Parts of the code are copyright Vladimir Arnost, Luc Cluitmans, Sebastien Bacquet.

WinTex code authors

Olivier Montanuy (montanuy@inf.enst.fr)	Most of WinTex code.
Vladimir Arnost (XARNOS00@dcse.fee.vutbr.cz)	MUS player.
Sebastien Bacquet (Sebastien.Bacquet@iie.cnam.fr)	MUS to MIDI converter.
Luc Cluitmans (L.J.M.Cluitmans@ele.tue.nl)	Hexen behavior decompiler.

And let's not forget **Serge Smirnov,** for the CleanWAD external tool!

WinTex coding help

Matt Fell	DOOM specs
Mark Mathews	Code portability
Elias Papavassilopoulos and Steve McCrea	Misc. tricks
Denis Moeller	Source of motivation :)
Robert Fenske, Jr and **Jens Hykkelbjerg**	(reject and nodes)

People who helped designing WinTex

Avery Andrews (tutorial on texture and flats).
Drake O'brien (tutorial complements).
Graham Evans (tutorial on sprites).
Alexander Prack (usage of custom flats).
Valery Tsyplenkov (critical technical help).
Kevin McGrail (HevKev) (help file of DeuTex).
Renaud Paquay (technical help).
Sebastien Bacquet (bug tracking and design help).

Support of WinTex

Avery Andrew, Steve McCrea, **Jens Hykkelbjerg, Marc-Andre Bissonnette**,**Jean Francois Fortin**, **Dan Teeter**,**Tony Muray, John Anderson (Dr Sleep)**, **Colin Caird, Sebastien Bacquet**, and many others that kept sending me help, support and bug reports by mail or by e-mail.
So many that I can't possibly remember then all!

Last, **thanks to you**, dear user of WinTex!
May the bugs of WinTex be kind to you and to your WAD files!

Informations about WinTex

First, the name: **WinTex** is named that way because it's a windows version of **DeuTex**, my previous freeware tool. DeuTex was named that way because it was first derived from **DEU** (like many other DOOM tools) and originaly dealt with Textures only.
Thus, the name **WinTex** is now a complete nonsense. Well who cares about the name?
The clarity and stability of the code is all I bother about!

Visual Basic code

The interface of **WinTex**, named **WINTEX.EXE**, was made with **Visual Basic** 3.0 for Windows 3.1.
Visual Basic allows very fast prototyping, and sometime fast development... well, that is, when it's misfeatures don't require two weeks of additional coding.

Future versions of this WinTex should run without **Visual Basic**, because it loads the panels too slowly, and use too much memory.

C code

All the serious work is in fact coded in **C**.
The library **LBWINTEX.DLL** contains all the compiled C code.
Using **C** rather than **Visual Basic** made this tool smaller, less prone to crashes (surprising, no?) and of course much faster. Also, object-oriented techniques have been used in the C code, despite the fact that **C** is not object oriented (I don't trust C++ and never will).

External Tools

WinTex 4.0 does not rely on my **DeuTex** tool for DOS anymore.
This is the main reason why it is so much easier to use than **WinTex** 3.41.

WinTex still uses my **DeuSF** tool, because I felt that operations like Merge, Join and App were safer in **DOS** than in **Window**, because those operations modify the **main WAD**.
WinTex by itself will never modify the main WAD.

Also, **DeuSF** has to be distributed with **WinTex**, since it is used to install some special WADs.

Installation and Setup of WinTex

Installation Steps

1/ Install WinTex on your computer.

2/ Configure WinTex for your favourite Bitmap, Wave and Text editor.

3/ Configure WinTex for use with your favourite level editor.

4/ Configure WinTex for use with your favourite WAD tools.

Additional Informations

Most of the DOOM related tools are not designed for Windows, so it will be a bit more complex.
But you may leave this for later: When you will need an external tool, WinTex will prompt you for the name of the executable.

In case of problem you can directly edit the WinTex.INI settings.

If you're a beginner, you probably should probably let this wait.

Installation of WinTex

Note: the file WINTEX.INI contains default settings that may not be suitable for your system. Delete this file if it causes troubles.

1. Make sure you have VBRUN 300.DLL

This is a file all **Visual Basic** 3.0 program need.
If you already used a Visual Basic program, you probably have it. Otherwise, it's available on Internet (ftp.microsoft.com), Compuserve, AOL, Prodigy or every software BBS.

2. Create a directory for WinTex and put all the WinTex files there

This directory must be readable and writable, so it cannot be on CD-ROM.
This directory should contain:
wintex.exe, the VB executable, that makes WinTex so cool (and so slow with windows).
lbwintex.dll, the C code that makes WinTex so fast with WAD stuff.
threed.vbx, some ugly code used for so-called 3D graphic controls.
deusf.exe, a tool WinTex sometime needs (it's a small version of DeuTex).
wintex.ini, an automatically generate file that contains various settings for WinTex.

3. Create a temporary directory on your system

WinTex needs to store some temporary files to disk, to exchange data with other tools.
Having a temporary directory is a good way to avoid making a mess in the WinTex directory.
Your temporary directory should be **C:\TEMP** or **C:\TMP**. WinTex will detect if the environement variable **TEMP** or **TMP** is set, and act accordingly.
If that doesn't work, then from withing WinTex, in the menu **Option**, click **Temp. Directory** and
select the directory of your choice.
You can use the WinTex main directory, if you don't have a better choice.

4. Launch WinTex

Just double-click on **WinTex.exe**.

It's possible, but not recommended, to launch more than one WinTex. If you must open more than one WAD, don't forget that WinTex can handle up to 8 WADs at the time.

5. Select a game

WinTex can work with **DOOM**, **DOOM2**, **UDOOM**, **HERETIC**, **HEXEN**.
However, you can only use one of these games at the time, and you must tell WinTex what game you are using. DOOM and UDOOM are considered the same.

Assume you want to use WinTex on **DOOM2**.
When you start WinTex, the WinTex **main window** appears. You will notice a panel named **main IWAD file** in the upper part of the window.

This panel contains a few buttons. Click on button **DOOM2**.

WinTex will ask you the name of the directory containing **DOOM2.WAD** and **DOOM2.EXE**.

For instance, it could be **C:\DOOM2**. If you give an invalid name, it will be ignored.

Changing WinTex default settings (INI file)

Contentes of   wintex.ini

The WinTex .INI file contains all sort of information that are specific to WinTex. It is not recommended to fool around with this, because I'm not sure WinTex will appreciate.

However, there are a few section that store the name and command lines of the Tools used by WinTex. These command lines are used to send precise informations to the tools.
Do not fool around with them, unless you know what you're doing.

[Tool]	contains the command line of the basic tools
[LevelEdit]	contains command lines for the DOOM level editors
[RejectEdit]	contains command lines for the level reject data builders
[NodeEdit]	contains command lines for the level node data builders
[ExeEdit]	contains command lines of DeHacked and HHE, the .EXE editors.

Explanations of all placeholders in the command lines

The command lines are built with placeholders (or, variables), that represent names that are subject to change. For instance %WAD% represent the name of the WAD file edited, and of course it changes each time a given tool is called.

Placeholder	Represents	Exemples
%IWAD%	.WAD file, main WAD of the game.	doom2.wad heretic.wad
%GAME%	Name of the game	DOOM2 HERETIC
%WAD%	.WAD file edited, containing just one level in.	e3m2.wad trinity.wad
%RWAD%	same, but without .wad extension.	e3m2 trinity
%LEVEL%	Level edited	e3m2 map01
%TEXU%	.WAD file, containing custom textures or flats.	texture.wad flat.wad custom.wad
%OLD%	.OLD file, that was previously a .WAD.	e2m3.old
%WADED%	WadED game indication	-1 (for DOOM) -h (for HERETIC)
%EDMAP%	Edmap level indication	/M:E2M3 (for E2M3) /M:01 (for MAP01)

Setup of basic editing tools (BMP, WAV, TXT)

For each basic type of file, WinTex will ask you what tool you want to use to edit it.
See the table below for the list of basic file types and suggested tool.

Select the executable of your editing tool

WinTex will interrogate **WIN.INI**, section [Extensions], to know what are your **favourite** Text, Bitmap and Wav editors.

If WinTex finds such a tool, it will ask you if it can use it.

If you reply no, WinTex will let you select the your favourite tool (point to the **.exe** file).

Command Line

Then WinTex will ask you what is the command line for your tool.
By default, WinTex assumes the command line is **%FILE%**. Don't be scared by this strange **%FILE%** name. It's only a **placeholder** for the name of the real file that will be edited with the tool.

If your tool is a **standard** windows tool, like pbrush, mplayer, winword,... **just press OK**.
If your tool take some special option in the command line, just write this option.
For instance, if the the tool needs the option **-file**, then write **-file %FILE%** for the command line.

Suggested Tools

TXT	text files, in DOS or Unix formats.	**UltraEdit** (a shareware) Any plain text editor, unlimited in size Notepad.exe (if you have nothing else)
BMP	bitmaps, 256 color, not compressed.	**PSP31.EXE** (Paint Shop Pro 3.1, a shareware) Pbrush.exe (if you have nothing else)
WAV	wave sounds, mono, 8 bits, not compressed.	**WHAM.EXE** (Wave Hold And Modify, a shareware) Soundrec.exe (if you have nothing else)
LMP	lump of data.	**PV.EXE** (ProView for Windows, a shareware) Any Hex editor will do ;-)
SCR	DOOM end screen data.	**TED.EXE** (TED12.ZIP on all DOOM ftp sites) ENDOOM.EXE (ENDOOM.ZIP on all DOOM ftp sites)
MUS	DOOM Music.	**MusPlayer** (MUSPL165.ARJ on all DOOM ftp sites) MUS2MIDI.EXE (to convert to midi)
MID	MIDI Music.	Any MIDI player, better than MPLAYER.EXE.

Setup of Level Editors for use with WinTex

Wintex is not a level editor: rather, it is meant to be used as a complement to existing level editors.

When you view a level map with WinTex, you can call your favourite editor on that level, modify anything you want, then reload it into the WAD file edited by WinTex.

Selecting your favourite level editor

In the **Level Editing panel** of WinTex, in the **Level Edit menu**, there is a sub-menu where you can select the editor you intend to use.
When you will first try to use this editor, WinTex will ask you to point to the executable (.exe) of this level editor.
For example, with **WinDEU**, you will point to **WINDEU.EXE**.

Particularities of your favourite level editor

There are many level editors around, each with it's own particularities. Most have not been designed to be called by an external tool like WinTex.
WinTex tries to cope with the most common ones (at least the versions I know about), by using either a specific command line, or, for the windows-based editors, by writing a specific .INI file.

Despite I checked those command lines, you may have to tweak a bit with those, if the editor doesn't work the way it should.
This command line contains options, like **-file**, **-main**, **-w**, and also strange placeholdders, that are filled by WinTex when it calls the editor.

Example of command line: WinDEU

The command line for WinDEU (or DEU) under WinTex is:

windeu **-w %IWAD% -file %TEXU% %WAD%**

This means that when calling WinDEU, WinTex will send something like:

windeu **-w doom.wad -file texture.wad   e2m3.wad**

More information about the placeholders, see the format of the WinTex INI File.

Recommended level editors

It's very difficult for me to recommend DOOM editors, because they all have lots of qualities, but also lots of bugs. The good editor is the one you manage to use.
That's why WinTex attempts to support so many of them.

Good editors include, in no special order:

DOS:	DEU 5.3 (not 5.21), DCK, EdMap (fast), DeeP/HeeP, WadEd.
Windows:	WinDEU, DoomCAD (slow on 486), EgoEd (slow on 486!), WadAuthor

All these programs are available on the DOOM ftp site (ftp.idsoftware.com, ftp.cdrom.com and mirrors), on CompuServe ACTION forum, and on AOL.

Setup of WAD tools for use with WinTex

Select your favourite tools

WinTex doesn't include a BSP node and Reject map builder.
It will never contain a .EXE editor, because this is not legal to modify game executables.

Instead, it let you use your favourite BSP node and Reject map builders (there are many of them) and your favourite .EXE editor (well, actually, there is only one: **DeHacked/HHE** by Greg Lewis).

Node and Reject builder

The first time you use any such tool, WinTex will ask you where the executable lies.
Then it will propose a command line for the tool. This command line may not be right, because there are so many tools and so many versions, so you will have to tweak a bit.

DeHacked

One exception is DeHacked: WinTex only supports DeHacked and HHE versions above 2.3, because you don't want to use the versions below, they have bugs ;-)
DeHacked 3.0 or more is recommended, since even version 2.3 still had some bugs.

Example of command line: RMB 2.1

The command line for Reject Map Builder under WinTex is:

rmb **%OLD% %WAD% NOMAP**

This means that when calling RMB, WinTex will send something like:

rmb **e2m3.old e2m3.wad NOMAP**

More information about the placeholders, see the format of the WinTex INI File.

Recommended tools

This is my own opinion only. For BSPs, it's based on examination of the code, when available.
For the other tools, it's based on more remote knowledge of the way they work.
There are some tools that I may not know about, so don't trust this too much.

Node builder	Good:	Warm 1.5, DEU 5.3, ZenNode
	Fast and Fair:	DeepBSP (from DeeP)
	Fair:	ID BSP
	Not good:	BSP 1.2, DEU 5.21, DCK 2.3
Reject builder	Best:	RMB 2.1a
	Fast:	Warm 1.5
	Not good:	Reject
EXE editor	Doom, Doom2:	DeHacked 3.0

Heretic:	HHE 2.3

All these programs are available on the DOOM ftp site (ftp.idsoftware.com, ftp.cdrom.com and mirrors), on CompuServe ACTION forum, and on AOL.

Browsing WADs with WinTex

You want to take a look at the contents of a WAD file? Easy!

1. Select a game

From the **main** WinTex window:
In the panel **Main IWAD File** choose the game you are interested in.
For instance, for DOOM2, press on button **Doom2**.

2. Select the WAD file to browse

From the **main** WinTex window:
In the panel **Pwad to examine or modify**, select the relevant directory, then click on a WAD file in the list. Only Wad files are listed. Then click on button **Browse PWAD** on the left.
The WinTex **wad editing window** pops-up. You can view the WAD, but not modify it.

3. View ordinary entries

In the WinTex **wad editing window**, just click on the entry of your choice.
WinTex will display it as *text* or *picture*, or *play it* on your soundcard.

4. View Wall Textures

Here is how to view the List of Textures defined in the WAD file.

5. View level maps, BSP nodes and REJECT data.

In the WinTex **wad editing window**, click on menu **Advanced | Edit levels**.
The WinTex **level editing window** will pop-up.

If your WAD contains levels, you will see those level appear in the list at the left.
It might contain names like MAP01 or E1M2.

Click on one of these levels. The level layout will appear.
Click on the **button Node**, the nodes of the BSP tree will be displayed.
Click on the **button Reject**, the reject data will be displayed.
Click on the **button Sides**, the wall sides in the level map will be displayed.

Right Click to *zoom in*, press **Shift** and **Right Click** to *zoom out*.

Maybe now you want to know how to modify the levels.

Viewing the Textures

In the WinTex **wad editing window**, click on menu **Advanced | Edit Texture**.
The WinTex **texture editing window** will pop-up.

This window contains a list of textures, on the lower left part.
Click on any texture in the **list of textures**, and it will be displayed.

Maybe now you want to know how to modify the textures.

Viewing the nodes of the level BSP tree

In the WinTex **wad editing window**, click on menu **Advanced | Edit Levels**.
From the **Wintex level editing window**, select a level and press on **button Node**.

One **node** is examined at the time:
- The node split line is displayed in **Yellow**,
- The right branch is displayed in **Green**,
- The left branch is displayed in **Red**.
- The part of the tree that is out of the node is displayed in **Grey**.

If you click in the **Green** part, you jump to the **Right child** node.
If you click in the **Red** part, you jump to the **Left child** node.
If you click in the **Grey** part, you go back in the node tree.

Note: If you don't know what a BSP tree is, just click around and it will soon become fairly intuitive for you. BSP is a short of Binary Space Partition Tree, a structure that makes the rendering of ***hidden*** *faces much* ***faster****.*

See here how to Build correct nodes for your level.

Viewing the level reject data

In the WinTex **wad editing window**, click on menu **Advanced | Edit Levels**.
From the **Wintex level editing window**, select a level and press on **button Reject**.

Click on any part of the map, to select one sector.
Imagine this is the sector where the player is running.
The other sectors are the sectors where monsters are waiting in ambush.

The sector where the player is will appear in **Yellow**.
The sectors whence monsters **can see** the player will appear in **Purple**.
The sectors whence monsters **cannot see** the player will appear in **Grey**.

See here how to Build correct reject data for your level.
See here how you can make Special effects by using the reject data.

Viewing the Sides

In the WinTex **wad editing window**, click on menu **Advanced | Edit Levels**.
From the **Wintex level editing window**, select a level and press on **button Side**.

Click on a **level** from the level list.
The level will be displayed.

Then Click anywhere on the level map.
The **closest facing side** will be displayed.
It will be **Green** if it is a right side (normal side)
It will be **Red** if it is a left side (back side of a two sided line)

Oh, I forgot to say: this function is **not useful** at all!
It will be useful in future version, for the level texture editing function.

Combining WADs with WinTex

Menu

Join two WAD files into one.

Add all sprites to your WAD (**required** for a WAD with custom sprites).

.

Add all flats to your WAD (**obsolete** function)

Modify the game's main WAD file by adding a WAD file inside.

Combining two WAD files into one

Suppose for instance that you want to join the WAD files TRINITY.WAD and VOIDSHIP.WAD into one single WAD file, TRINITY.WAD.

Since both those famous WAD use redefined textures, ordinary tools will not be able to join them into one WAD. Hopefully, the tool **DeuSF** has a special trick to deal with such cases.

Usage

From the WinTex **main window**, select a **game**, for instance **DOOM2**.

You will notice a panel in the lower part of the window, that is named **'WAD to examine or modify'**. Go there and select a WAD file, for instance **TRINITY.WAD**.

You will notice a panel in the middle of the window, that is named **'WAD to Join'**.
Go there and select another WAD file, for instance **VOIDSHIP.WAD**.

The just press on the button **Join PWADs**.

DeuSF will run for a while, and append **VOIDSHIP.WAD** to **TRINITY.WAD**.

This is **reversible**: just select TRINITY.WAD and click on button **Restore PWAD**, and TRINITY.WAD will be back to normal. Now talk of professional tools!

Important

In case two entries are conflicting, WinTex will **discard** the entry from the WAD that is being **modified**.
for instance, if both WAD files redefine level **MAP01**, then the level **MAP01** from the modified file will be discarded.

Using DOOM WADs with Ultimate DOOM

An unexpected consequence of the WAD combining capability of **DeuSF** (coded long before Ultimate DOOM was out) is that if can be used to use DOOM wads with Ultimate DOOM.

If you have **TRINITY.WAD**, and the **DOOM.WAD** from Ultimate DOOM, just **join** DOOM.WAD with TRINITY.WAD.

Then DOOM.WAD will contain TRINITY.WAD, and the textures will be re-calculated, so that it can be played without problems.

Adding all sprites to a WAD file

Suppose you have a PWAD file that contains some **sprites** that were modified. You may expect such a WAD to work like any usual PWAD, but this isn't quite true. If you try to play this WAD (using -file) the game will **lock up**. This is a bug of DOOM.

There are some tricky way to overcome that bug, by modifying the game executable, but the easiest way is to use the function of **DeuSF** that adds all sprites to a PWAD.

When using the **Add all sprites function**, you put in the WAD all the missing sprites, and then DOOM (or DOOM2) will accept to **run** the WAD. Don't try to understand why!
(Note: it works also with HERETIC, but it's less reliable: prefer the **Merge** function.)
(Note: if a sprite WAD was **made for DOOM**, then adding all sprites of DOOM2 will enable it to **run under DOOM2**.)

The big **drawback** is that it takes a lot of time, because it adds about 4 meg of data to your WAD, in the worse case.
Of course, do not distribute a WAD with all flats appended to it! If you really want to distribute such a WAD, learn how to install it.

Usage

From the WinTex **main window**, select a **game**, for instance **DOOM2**.

You will notice a panel in the lower part of the window, that is named '**WAD to examine or modify'**.   Go there and select a WAD file that contains custom sprites, for instance **SIMPSON.WAD**.

The just press on the button **Add all sprites**

DeuSF will run for a while, and append all sprites of **DOOM2.WAD** to **SIMPSON.WAD**.

This is reversible: just select **SIMPSON.WAD** and click on button **Restore PWAD**.
Then WinTex will tell you the WAD has been restored, and **SIMPSON.WAD** will be back to normal.

Maybe now you want to know how to modify the sprites.

Adding all flats to a WAD file

This function was required in former versions of the game engine, because they could not handle modified flats present in PWAD files.

If you built your PWAD file with a recent version of WinTex, then **you don't need** that function.
WinTex now handles the flats automatically, and you should not bother about it anymore.

Maybe now you want to know how to modify the flats.

Modifying old WAD files containing custom flats

Suppose you have a PWAD file that contains some **flats** that were modified.
If you wish to be able to use it without problems, just open the WAD with WinTex.

Browse the WAD, and make sure that at the end of the list of flats, there is an entry called **F_END**. If there is one, then your WAD is usable by the game.

If there is an entry called **FF_END**, then under the WinTex **Wad editing window**, click on menu **Guru | Use F_END**. This will transform FF_END into F_END.

If there is an entry called F_START at the begining of the list of flats, then the game will suppose that the complete list of flats will be provided between F_START and F_END.
So remove that F_START entry, because the complete list of flats is only in the main WAD.

Usage of the (obsolete) Add flats function

The Add flats function, though obsolete, was not removed from WinTex because it might still be usefull (if you own an old version of the game engine, or are encountering some troubles with flats).

From the WinTex **main window**, select a **game**, for instance **DOOM2**.

You will notice a panel in the lower part of the window, that is named '**WAD to examine or modify'**. Go there and select a WAD file, for instance **TRINITY.WAD**.

The just press on the button **Add all flats**

DeuSF will run for a while, and append all flats of **DOOM2.WAD** to **TRINITY.WAD**.

This is reversible: just select **TRINITY.WAD** and click on button **Restore PWAD**.
Then WinTex will tell you the WAD has been restored, and **TRINITY.WAD** will be back to normal.

Modifying the main WAD of the game

WinTex allows you to modify the game's main WAD file (for instance, DOOM2.WAD) in a very secure and very reliable way.

Note: The method is completely different from the method used in other tools, and only a minimal modification is made to your main WAD file. When this main WAD file is later restored, this is a complete and perfect restoration, meaning that you can even use upgrade patch on this main WAD. The method is much better than the one used in DMGRAPH for instance.

If you don't want to modify the main WAD, check how to modify the WAD files.

Why modify the main WAD?

There is one good reason for that: **Bugs**.
Not bugs in WinTex, bugs in **DOOM**, **DOOM2** and **HERETIC**.
You know of course that you can play PWAD files by using the **-file** parameter, and the name of the PWAD. but due to the lousy file system used by ID software, this **-file** cannot handle properly PWADs that contain modified sprites and flats.
Merging a PWAD withing the main WAD is the only way to get rid of all the restrictions.

There is another good reason: **Network play**.
I have never managed, personnally, to run a fine network game with the **-file** parameter. There seem to be another bug there, and you will get more game play inconsistency that you should.

There is a last reason: **Portability**
Sprite and flat PWADs meant for DOOM can be merged into DOOM2.WAD, and they will play just fine. (That was not possible with tools like DMADDS!)

Usage

Suppose for instance that you want to modify **DOOM2**, so that the sprites from **SIMPSON.WAD** are always used by **DOOM2**.

From the WinTex **main window**, select a game, for instance **DOOM2**.

You will notice a panel in the middle of the window, that is named **'WAD to Join'**.
Go there and select another WAD file, for instance **SIMPSON.WAD**.

The just press on the button **Merge in IWAD**.

DeuSF will run for a while, and append **SIMPSON.WAD** to **DOOM2.WAD**.

This is reversible: just click on button **Restore IWAD**.
WinTex will tell you that the main WAD was restored, and **DOOM2.WAD** will be back to normal.

Editing WADs with WinTex

General modifications

How to reuse entries from other WADs.
Modify the data lumps (texts, screens, demo, colors).
Modify the pictures (title pictures, status bar, helps).
Modify the hexen special data (scripts, map defs, sounds defs, animations).

Modify monsters data

Modify the sound effects (wave and PC speaker sounds).
Modify the sprites (monsters, weapons).

Modify level data

Modify the wall textures (textures, sky, animations).
Modifythe wall patches (parts of textures).
Modify flats (floors and ceilings).
Modify musics (level musics).

Technical informations

How WinTex modifies WAD files.
How to save the WAD directory.

How to clean the WAD file

When you have finished editing a WAD with WinTex, please clean the WAD before distributing it to anyone.

Tutorials

Tutorial on WAD installation.
Tutorial on Pictures editing.
Tutorial on Sprites editing.
Tutorial on Level editing.
Tutorial on Textures editing.
Tutorial on Floors and Ceilings editing.

Cleaning WAD files

When WinTex edits a WAD file, it doesn't optimise the organisation of data in the WAD file.
- Optimising the organisation of a file takes too much time (file I/O is so slow...)
- If WinTex deleted unused data, then there would be now way to cancel changes.

This is not specific to WinTex: most WAD editors proceed that way (and even some text editors like MS-WORD).

But this means your WAD file will keep growing in size, and a lot of disk space will be wasted.

This is not a problem: if your WAD gets too fat, just clean it.

Fast cleanup

Under the WinTex **Wad editing window**, click on menu **File | Cleanup WAD**.

> This WinTex cleanup function is relatively fast, but does not make any real optimisation.
> The previous WAD file is preserved, and renamed with extension .old.

Optimised cleanup

Under the WinTex **main window** click on menu **File | Clean WAD**.

> This will invoke the external tool **CleanWad**, a very usefull freeware by **Serge Smirnov**.
> CleanWad will try to reduce your WAD file down to the minimum size.
> Use this tool just **before you distribute** your WAD file.

How WinTex modifies the edited WADs

Entries in WAD files

Each time you read a new WAD entry, or modify it, or import it form another file, WinTex keeps a copy of both the old entry and the new one. WinTex never deletes any entry.

Since the WAD containes more data than is really needed, it is bigger than it should.
The WAD file thus contains data separated, apparently, by hole: lots of unused data that seems wasted.

This does not cause any problem when editing or playing the game. It's even quite convenient because you can cancel the last changes made to your WAD.
But don't forget to clean the WAD before distributing it.

WAD directory

WinTex also keep a copy of the previous WAD directory, together with some informations about the previous file size, various pointers and even the time stamp of the file.

This allows WinTex to make complete restorations.

A Wad modified by **WinTex** can be restored by **DeuSF** or **DeuTex**, not by **NWT**.
The other tools don't have any WAD restoration feature.

WAD restoration (pseudo undo command)

The WAD restoration function brings you back to the situation just before you started your WAD editing session. It cancels all the modifications made during that session.

Don't confuse this with a **undo** command: it will undo all the modifications, not only the last one. There is no undo command because it makes the code too complicated (and unsafe).

Note:
If you restore a WAD, you will lose last your modifications forever, but WinTex will warn you.

Saving the WAD directory

The WAD directory is the list of contents of the WAD file.
It's basically the list you see when you click on menu **Select | All**.

WinTex automatically updates the WAD directory.

However, if you intend to use another editor on the WAD, in the same time as WinTex, you should ensure manually the WAD directory is updated:
From the WinTex Wad editing window, click on menu **File | Update Wad**.

Changing the default transparent color

What is the transparent color?

When you draw a monster picture, you usually want that the area around the sprite remains transparent. So you give it a special color, the transparent color.

The usage is to use the color **cyan** as transparent color, because this color is seldom used in realistic scenes. However some images, especially those generated by 3D Studio or POVRAY, might have a **black** background.

So WinTex let you modify the color that is considered transparent.

How WinTex determines the transparent areas?

WinTex will compare the color of each pixel with the reference transparent color.
If they are **strictly equal**, then WinTex will consider the pixel transparent.

Warning: If the color is **cyan** (Red=0, Green=255, Blue=255) but you use a color close to cyan (Red=0, Green=252, Blue=252), then WinTex will not consider your color transparent.

Change the transparent color

From the WinTex main window, click on menu **Option | Transparent color**.

WinTex gives you the choice between three default colors, and a custom color.

- **Cyan** is the default color, used by every tool since DMGRAPH.
  Cyan is used because it doesn't exist in the game, and is a very uncommon color in real-life pictures, except in the sky.

- **Black** is the color to use, if you generated sprites with POVRAY.
  POVRAY and 3DS draw pictures on a black background, by default.

- **Dark Grey** is the favourite color to use for the **background of sprites**.
  That's because if you smooth a picture on a dark grey background, then its edges will look fine in the game. Don't use black or cyan for that!.

Once you set a transparent color, it is used in every current or later WinTex sessions.

The picture format

WinTex currently accepts only Bitmap pictures, because it's the most common format under Windows, it's easy and fast to read, and reliable.

Here are the required settings for your Bitmap editors:

MS Windows Bmp format, not the OS/2 variant.
The format must be strictly respected.
Use the DOOM, HERETIC or HEXEN default color palette.
No compression: RGB encoding, not RLE encoding.
256 colors bitmaps (8 bit per color).

Conversion of Bitmap format

If you are using 24 bit (Million colors) bitmaps, WinTex will take them but the conversion will be **extremely slow** and **inadequate** (no dithering).

Conversion of colors

If your picture is not using the right palette, WinTex will automatically find the closest colors in the game palette. Since there is no dithering, the result might not be satisfying.

So it is highly recommended that you proceed to color conversion from within your Bitmap editor. Most good editors will let you save the palette of a reference game picture, and later upload it into one of your pictures. Neopaint and PSP 3.1 for instance.

Strict respect of formats

The Bmp format is quite lousy, and some badly coded tools do not respect it entirely, but WinTex will attempt to compensate.

Future version of WinTex may support TGA, PCX, GIF, or rather the new Compuserve format.
If you need that format, ask! I may take the time to code it.

Cut and Paste entries from different WADs

If you want to move entries between WAD files, the easiest way is to use the Copy Cut and Paste functions.

Copying entries from a WAD

Open a WAD file with WinTex.
In the WinTex **wad editing window**, select the entries you wish in the entry list.

You can use Shift-Click to select a group of entries, or Control-Click to select one by one.

Then click on menu **Edit | Copy entries**.

Paste entries into a WAD

Open a new WAD file with WinTex.
In the WinTex **wad editing window**, select a position in the entry list, where you want the entries to be inserted.

Then click on menu **Edit | Paste entries**.

Warning: you will sometime notice that the entries are not pasted where you wanted. This is because you chose an invalid position for that kind of entry, and WinTex did the necessary adjustments.

Modifying the data lumps (demo, texts, misc.)

Modify a lump

From the WinTex Wad editing window, click on menu **Select | Lumps**.

Double click on any listed lump to edit that lump.
Click on menu **Edit | Load entry from file** to load a new lump.

Particularities of lump entries

There are many kind of lump entries, each with it's **own** particular format.
It you change the data associated to such an entry, the new data must have the exact **same format** as the old data.

WinTex knows the most common type of lumps, and will handle the data you provide depending on the kind of lump.
But if your lump is of a very special kind, and WinTex doesn't know the format, then WinTex will just load that lump as a raw data file, without even bothering about the format. So you'd better know what you are doing.

Common lump formats

Type of lump	**Format expected**	**Example of entry**
DEMO	Special game demo format (See the specifications)	DEMO1

SCREEN	Text screen 80x25, colored.
	(like the DOS text mode)
		ENDOOM
		ENDTXT

TEXT	Text configuration file	DMXGUS MAPINFO
BITMAP	320x200 flat picture (Only for Hexen/Heretic)	TITLEPIC
PALETTE	256 RGB colors	PLAYPAL
COLORMAP	Color mappings	COLORMAP FOGMAP
RAW DATA	format unknown	GENMIDI TINTAB

Modify the data lumps

The text or picture data are editable easily.
The demo can be recorded by using the game. They cannot be modified.

The other data can probably be edited, but you will need some special external tool.

Hexen special data

See how to modify the hexen lumps (MAPINFO, SOUNDINFO, ANIMDEFS...)

Modifying the sound effects

From the WinTex Wad editing window, click on menu **Select | Sounds**.

Double click on any listed entry to edit that sound.
Click on menu **Edit | Load entry from file** to load a new sound.

Usage of sound entries

The sounds represent the various **sound effects** that are played when for instace a weapon is used, or when a monster shouts, or when a door closes.

The **name** of the entry completely **determines** what the sound will be used for, and this is hard coded in the game executable.

So to **replace** a sound in the game, you must simply put in your WAD a sound entry of the same name.

When you select a sound entry, WinTex tells you what it is actually **used** for.
For instance, WinTex will tell you that **DMPISTOL** is the name of the pistol sound.

WinTex will also automatically start playing the sound. If you **don't want to hear the sound playing**, click on menu **Select | No Sounds**.

Particularities of sound entries

The only format of sound accepted by WinTex is a very simple one:
WAVE files (file with extension .WAV)
11025 sample per seconds (prefered rate, but you can use others)
8 bit per sample (not 16 bit per sample)
No compression (PCM encoding, not ADPCM)

The sound can be up to 64K in length. If you provide more than 64K, the game will ignore the data after 64k. (So if you want long sounds, use a slower sample rate.)

Hexen sound names

See how to modify the hexen sounds names (SOUNDINFO)

Modifying the wall textures

Don't forget to read the tutorials.on texture editing.

Modify a texture

From the WinTex **Wad editing window**, click on menu **Advanced | Edit Texture**.
This will start the WinTex **Texture editing window**.

On the **right** of the WinTex **Texture editing window**, you can see the list of patches.
Click on menu **Patches | List Patches** to obtain the list of all the wall patches (those inthe main WAD but also the custom patches in your PWAD.)
Alternatively click on menu **Patches | List Sprites** to obtain the list of sprites, because you can also use sprites in textures.

Click on any patch in the list, it will appear on the texture.

Position the patch on the texture, click on it, and it will be dropped there.

In the **center** of the WinTex **Texture editing window**, you can see the list of patches in the texture currently edited. You can move or delete patches in that list.

On the **left** of the WinTex **Texture editing window**, you can see the list of textures.
When you create a new texture, it will be inserted above the currently selected one.

Usage of the textures

The textures are the pictures that appear on the level walls (not on the floor or ceilings).
They are not actual pictures, but rather are made of a combination of smaller pictures, the wall patches.

Special textures

Some texture names are special: they are used for animations, switches or sky.
WinTex will report that usage, if such a texture is selected.

How to create animated textures.
How to modify the textures of the switches.
How to modify the sky texture.

Use custom textures in levels

See how to use custom textures in your levels.

Copy textures from other WAD files

If you want to reuse textures from another WAD file, just open this WAD file, select all the textures in the texture list, copy those textures into the texture clipboard
From the WinTex **texture editing window**, click on menu **Textures | Copy**.

Then just paste those textures into the texture list of your new WAD:
From the WinTex **texture editing window**, click on menu **Textures | Paste**.

Warning: if you are using some **custom wall patches** in those textures, you'd better first cut and paste those patches into the new WAD file, otherwise WinTex will not find them.

Modifying an animated wall texture

Don't forget to read the generalities about texture editing.

What is an animated texture?

An animated texture is a texture that change according to a repeating pattern.
Such a texture is typically used to create wall of fires, but is also commonly used to create small animation films on the walls.

There is only a limited set of animation possibles, and the name of the animation are hard coded into the game executable. So you have to redefine an existing animation, you cannot just create a new animation.

How to create an animated texture?

Internally, an animated texture is in fact made of two or more textures, that must be consecutive in the list of textures, and that must be placed between two textures with special names: the start of animation texture, and end of animation texture.

To create animated textures, redefine the textures that are the start and the end of an animation sequence, and put some new textures between those start and end textures.

the whole lot of texture between the start and end of animation will then become animated.

Note that it is **critical** that your new textures appear in the list **after** the animation start, and **before** the animation end. If you don't do that, the game will crash.

Hexen animated textures

See how to modify the hexen animated textures names (ANIMDEFS).

Modifying a swicth texture

Don't forget to read the generalities about texture editing.

What is a switch texture?

A switch texture is a texture that changes when you press a switch on the wall.
It's a bit like an animated texture, except that the animation is controled by the game engine, and has only two stage: switch off and switch on.

There is only a limited set of switches possibles, and the name of the switch are hard coded into the game executable. So you have to redefine an existing switch, you cannot just create a new switches.

How to create a switch texture?

Internally, a switch texture is made of two textures, with special names.
Generally, the two switch textures will have similar names. The switch off texture will have a 1 in the name, like SWBROWN1, and the switch on texture will have a 2, like SWBROWN2, but this not always the case.

WinTex will report if the texture is a switch, and if it is on or off.

If you want to create a new kind of switch, it is recommended that you replace the switch off texture of an existing switch by your own texture, and similarly for the switch on texture.

Modifying a sky texture

Don't forget to read the generalities about texture editing.

What is a sky texture?

A sky texture is no more than the texture that is displayed on the sky.
As a matter of fact, countrary to floors and ceilings, the sky is a texture.

But this is a very special texture, with size 256x128 at least.
Countrary to ordinary textures, it is flipped hoizontally (left becomes right, and inversely).
So if you want to put letters or oriented stuff on a sky, first flip it horizontally.

The names and usage of the sky textures are hard coded in the game executable.
Typically, SKY1 will be used for the first few levels, SKY2 for the next few levels and so on.
there is no known way to change that.

How to create a sky texture?

Most of the time, you will create a sky texture like any ordinary texture, except that the size of the texture must be 256x128.

It is higly recommended that you use only a few patches in a sky texture. More than 4 patches is likely to crash the game.

It is also recommended that you use patches of size 256x245. Such patches are apparently too high for the texture, but if you put the upper part of the patch on the sky texture, then the lower part of the patch will help prevent an ugly display if for some reason the sky must be painted real far downward. Under Heretic, it is recommended to use that kind of patches.

The non-repeating sky texture.

You can create a sky texture that will not repeat north, east, south and west, like the default a 256x128 sky does. This is especially suitable to display a somewhat realistic countryside in the background.

It's all very simple: create a **1024x128** texture made of four consecutive 256x245 patches.

The fist patch, at offset **0,0** will be the northern sky.
The second patch, at offset **256,0** will be the eastern sky.
The third patch, at offset **512,0** will be the southern sky.
The last one, at offset **768,0** will be the western sky.

You cannot use more than 1024 for the width of a sky.

Hexen sky names

See how to modify the hexen sky names (MAPINFO)

Modifying the general pictures

Don't forget to read the tutorial on picture editing.

Modifying a picture

From the WinTex Wad editing window, click on menu **Select | Pictures**.

Double click on any listed entry to edit that picture.
Click on menu **Edit | Load entry from file** to load a new picture.

If you have **troubles** with a graphic picture, check the picture format.

Usage of picture entries

The pictures represent the title pictures, help pictures, menu items, end level pictures.
The name of the entry completely determines what the picture will be used for, and this is hard coded in the game executable.

So to replace a picture in the game, you must simply put in your WAD a picture entry of the same name.
For most pictures, WinTex will tell you what the picture name means.
For instance, WinTex will tell you that **M_DOOM** represents the **DOOM** logo.

Particularities of picture entries

If you want to replace a picture by another one, please use exactly the **same** width and height as in the original picture.
For menu items, it is possible that using different heights cause no special trouble, but most of the time it should cause trouble when playing the game

Modifying the sprites

Don't forget to read the tutorial on sprite editing.
If you want to distribute sprite WADs, read the tutorial on WAD installation.

Modify a sprite

From the WinTex Wad editing window, click on menu **Select | Sprites**.

Double click on any listed entry to edit that sprite.
Click on menu **Edit | Load entry from file** to load a new sprite.

If you have **troubles** to load a sprite picture, check the picture format.

Usage of sprite entries

Sprites are the monsters, players and objects seen in the game.

The sprite entries are pictures that are used at different stage of the animation of sprites.
For one single sprite, there are many sprite entries, that will show the sprite walking, the sprite seen from front, from left, from behind, the sprite being hurt, or similar stuff.
Browse the list of sprites, and you will understand.

The name of the entry completely determines what sprite is concerned, what phase of the sprite the picture is used for, and what viewpoint it is used for.
This is hard coded in the game executable.
Of course, WinTex will decode the name for you, and write the result in the **viewpoint** panel.

Particularities of sprites

Modifying sprites is more difficult than it looks.
In particular, if a WAD contains modified sprites, it **cannot** be used by the game.

To be able to use such a WAD, for instance with **DOOM2**, you must either:

Append all the sprites of DOOM2
(use the menu **Merge | Add all sprites** from the WinTex main window)

Merge the WAD into DOOM2.WAD
(use the menu **Merge | Merge in IWAD** from the WinTex main window)

There is **no** other choice. This is due to a bug in DOOM2, that WinTex attempts to compensate, at the cost of some meg of wasted disk space.

The only way to avoid this is to use only new names for sprites, and use S_END as end sprite marker.

Sprite offsets

The panel on the upper right of the sprite picture show the offsets of the sprite.

Those offsets tell how the sprite picture should be placed, relatively to the point where the sprite is walking (this is the point where the two thin black lines cross).

For wielded weapons, the offsets position the weapon relatively to the screen (there is no cross displayed in this case).

You have the choice between 6 default offsets:

Weapon offsets: use it for most wielded weapon pictures.
Weapon offset 2: use it for some wielded weapons (if the first one doesn't work fine).
Walking offsets: use it for monsters that are walking.
Flying offsets: use it for explosions, and monsters that are flying.
Null offsets: use it only for graphics, not for sprites.

Then you can adjust the offsets like you wish.

Modifying the wall patches

Don't forget to read the tutorial on texture editing.

Modify a wall patch

From the WinTex Wad editing window, click on menu **Select | Patches**.

Double click on any listed entry to edit that patch.
Click on menu **Edit | Load entry from file** to load a new patch.

If you have **troubles** to load a wall patch, check the picture format.

Usage of patch entries

The patch entries are pictures, that meant to be used as components of wall textures.

You will build a wall texture from one, two, three or more wall patches that you will have previously put in your WAD. (In fact, you can also use sprites as wall patches.)

The name of the wall patches can be chosen at your convenience, there is no restriction, except that it must not conflict with another entry name (WinTex will tell you, then).

Particularities of wall patches

The size of the wall patches can be chosen at your convenience, but it must be smaller than 512x256, for reasons related to the encoding of pictures.

It is recommended that you use **64x128** or **128x128** for the wall patches.
Remember that ordinary textures pad every **128** in height, so using a height of more than 128 would be a waste, and using less than 128 would leave holes.

However, you may wish to use heights of more than 128 if your patch is meant to be used in a transparent texture, that will be put on a transparent or semi-transparent wall (a two sided wall).

Also, you may wish to use a height of 245, and a width of 256, if your patch is meant to be used in a sky textures.

Modifying the floors and ceilings

Don't forget the tutorial on floor and ceiling editing.
If you want to distribute flat WADs, read the tutorial on WAD installation.

Modify a floor or ceiling

From the WinTex Wad editing window, click on menu **Select | Flats**.

Double click on any listed entry to edit that flat.
Click on menu **Edit | Load entry from file** to load a new flat.

If you have **troubles** to load a flat picture, check the picture format.

Usage of flat entries

Flats are special **64x64** pictures that represent the **floors and ceilings** in the levels.

Note that the **sky** is not a Flat, it is a wall texture.

The flats can have any name you wish, provided it doesn't conflict with other names.
WinTex will tell you if a particular name cannot be used.

The Flat pictures

When you provide a picture that will become a flat, it must be of size 64x64, and the drawing must be designed so that it can smoothly tile vertically and horizontally.
This is because a flat is repeated every 64 X or Y coordinate, in a level.

Particularities of the flat entries

Modifying flats is more difficult than it looks.
In particular, if a WAD contains modified flats, it **cannot** be used by the game.

To be able to use such a WAD, for instance with **DOOM2**, you must either:

Append all the flats of DOOM2
(use the menu **Merge | Add all flats** from the WinTex main window)

Merge the WAD into DOOM2.WAD
(use the menu **Merge | Merge in IWAD** from the WinTex main window)

There is **no** other choice. This is due to a bug in DOOM2, that WinTex attempts to compensate, at the cost of some meg of wasted disk space.

Using custom flats with level editors

See how to use custom floor and ceilings in your levels.

Animated flats

When you put a flat, in the list of flats, between a flat that is the start of an animation, and a flat that is the end of an animation, then this flat becomes part of the animation.

WinTex will tell you the names of the flats that are the start of the end of an animation.

Don't put too much flats between the start and the end of an animation, otherwise the game may crash.

Hexen animated flats

See how to modify the hexen animated flats names (ANIMDEFS)

Modifying the musics

Modify a music

From the WinTex Wad editing window, click on menu **Select | Musics**.

Double click on any listed entry to edit that music.
Click on menu **Edit | Load entry from file** to load a new music.

This file must be under the **MUS** format.
If it's a MIDI file, the external conversion tool **MIDI2MUS** will be invoked.

Usage of music entries

There is one music entry associated to each level. This music is played continuously when you are in the level.

For instance, in **DOOM**, the music **D_E2M4** plays when you are in level **E2M4**.
In **DOOM2**, the name of the music is less easy to figure out, but WinTex will help you by providing the list of musics, and their associated level.

Particularities of music entries

The Music entries are not **MIDI** or **MOD** files. They are under a special format, **MUS**.

Hopefully, WinTex contains the QMUS2MIDI converter by **Sebastien Bacquet**.
So WinTex will save the files as **MIDI**, which is a more common format than MUS.

Playing MUS file under windows

Since WinTex contains the MUS player from Vladimir Arnost (Q&A software), it can play the MUS files under windows.

By default, each time you click on a music entry, WinTex will start playing the music.
If you want WinTex to keep quiet, click on menu **Music | No Sounds**.

If you just want to stop the music playing, click on menu **Music | Stop Music**.

Warning: due to some bug in the current MUS player, it will not work with a Gravis Ultrasound, or any soundcard that requires that instrument patches be loaded.
Then just **Double click** on any music entry, and WinTex will try to call your favourite **MIDI player**.

Modifying the HeXen special data

WinTex enable you to edit some of the data lumps of HeXen. Only minimal info will be provided here. For more information, please read the **Hexen Specifications** by Raven Software and **Ben Morris** (author of DCK and HCK).

Modify the level scripts (BEHAVIOR)

From the WinTex Wad editing window, click on any BEHAVIOR entry.
It will be displayed as text, thanks to the integrated DEACC decompiler by Luc Cluitmans.

You can save a Behavior entry as text, but Behavior is **not text**, but compiled data.
To modify it, you will need the ACC.EXE compiler provided by Raven Software.

WinTex does not provide any integrated compiling environement. Please use a HeXen level editor (WadAuthor for instance) because you will have to modify quite a lot of things on the level map to get your script running.

Modify the level list (MAPINFO)

This text data contains one set of definition per level in the WAD file.
Here is a commented example of a level definition:

```
map X   "NNNN"  ; X is the level id in WAD, and NNN the name of the level.
warptrans X     ; X is the cheat code number, to warp to the level
next X          ; X is the number of the next level in the sequence
cluster Y       ; Y is the id of the group of levels to which the level belongs.
sky1 TTT X                  ; TTT is the sky texture name, X is an horizontal offset
sky2 TTT X                  ; same.
doublesky                   ; self explanatory.
lightning       ; Means that there are lightnings in the sky.
fadetable EEE   ; EEE is the name of a WAD entry with color fading (fog effect).
cdtrack X       ; Track of the CD-Rom where the accompanying music is.
```

The text after **semicolon ";"** is a commentary.

Modify the sound list (SNDINFO)

This text data defines the WAD entries used for sounds and musics.

Example of sound definition:

```
NNN   EEE                         ; NNN is the long name of the sound
                        ; EEE is the name of a WAD entry containing a sound
```

The long names of sound are used in Scripts (see BEHAVIOR lump) and Platform sound definitions (see SNDSEQ lump).

Example of music definition:

```
$MAP   X EEE            ; X is the reference of the MAP in the WAD
                        ; EEE is the name of the WAD entry containing the music data
```

Modify the platform and door sounds   (SNDSEQ)

This text defines the sound associated to some animations.

Commented example of a definition:

```
:PPP                    ; PPP is the name of the animation/movement
    playtimes NNN X     ; play X times the sound NNN
    playuntildone NNN          ; play the sound NNN until movement is finished
    playstop NNN               ; play the sound NNN when movement stops
    playrepeat   NNN    ; play the sound NNN continuously
:end
```

Modify the animated textures and flats (ANIMDEFS)

This text data defines which textures and flats are animated, how many steps there are in the animations, and how long each of the animation steps last.

Example of definition:

```
texture TTT1                ; Animated texture, starts with name TTT1
pic   1   tics   4                 ; Step1: display texture TTT1 for 4 tics (4/35 seconds)
pic   2   rand   5   8             ; Step2: display texture TTT2 for 5 to 8 tics (random choice)
pic   3   tics   120               ; Step3: display texture TTT3 for 120 tics
pic   2   tics   4                 ; Step4: display texture TTT2.
                            ; the next step is Step1
```

Modifying the levels

Level maps

Modify the level layout.
Use custom textures in your levels.
Use custom floor and ceilings in your levels.

Note: WinTex cannot edit the level objects.

Level specials

Calculate the nodes of the BSP tree, before playing a level.
Calculate the reject data, to make the game faster.
Do special tricks, with the reject data.

Modifying the level maps

The level maps

The level maps are of course the places where you will be running around, as a player.

WinTex does not and will **never** edit DOOM or HERETIC levels.
There are lots of **good** DOOM editors around, and I consider it's a specialised job.
The current WinTex developments are aimed directly at Quake, which is much harder to edit.

So to modify the **level** maps, you will need a **level** editor. But WinTex will make it easy for you to integrate levels in one single WAD, and to fool around with those levels.

WinTex will save a temporary map file, feed it to your level editor, and then load it back when the editor has finished working on it.

You will of course need to configure configure that level editor for WinTex.
This is because all level editors have their personnal way to deal with WADs, and WinTex must tell them what game to use, what WAD to edit, and what is the name of WAD with custom the textures and flats.

Modify a level map

From the WinTex **Wad editing window**, click on menu **Advanced | Level editing**.
The WinTex **level editing window** will appear.

If your WAD contains levels, those levels will appear in the list of levels, on the left.

If not, you can still click on menu **Level | Load level**, give a name to your level, like **MAP01**, and then click on a WAD file containing a level you want to load in your WAD.
You can also cut and paste a whole level, from the WinTex **Wad editing window**, provided you open another WAD that contains levels. That might be more convenient.

If you want to modify textures in your level, WinTex can do the job.

But if you want to modify the level map, you need a level editor, for instance WinDEU.

Click on menu **Level | Editor** and select an **WinDEU**.
Then select a level in your WAD and click on menu **Level | Edit Level**, and WinTex will call **WinDEU** with the right parameters.

> WinTex will certainly ask you to point to **WINDEU.EXE**, and then WinTex will ask you to confirm the command line for WinDEU, this is just for security.

WinTex will then disappear temporarily, as WinDEU runs in the foreground.

Under **WinDEU**, modify your WAD as much as you want. Save your WAD to the same file name. Then exit WinDEU.
WinTex will then come back, and ask if you want to reload your modified level in the WAD.
Click **OK** if you were satisfied of your modifications, **no** otherwise.
Then WinTex will ask you if you want to delete the temporary file, click **OK**.

Calculating the nodes of the BSP tree

From the **Wintex level editing window**, select a level and press on **button Node**.

This calculation is very intensive, and is **critical** for a good game play:
- If nodes are build incorrectly, the game display will look fuzzy.
- If nodes builts are not optimal, the game will run slower than with good nodes.
(However, there are cases when even the best nodes cannot make the game faster.)

Click on menu **Node | Calculate**, that will save the level to a temporary file, and then run a node builder tool on this temporary file.

See the setup of Node builders tools.

Calculating the reject data

From the **Wintex level editing window**, select a level and press on **button Reject**.

There are many line of sight caculations needed to build even a basic reject data.
So you'd better first use an external tool on the level.

Click on menu **Reject | Calculate**, that will save the level to a temporary file, and then run a reject tool on this temporary file.

See the setup of Reject tools.

Special effects with the level reject data

From the **Wintex level editing window**, select a level and press on **button Reject**.

First check how the reject data is displayed.

Then check how the reject data can first be calculated automatically.

Modifying the reject data

Warning: *Edit reject data only when you are sure you will not need to add new sectors you your level. Otherwise, all your hard work will have to be discarded.*

Press **Shift** and **Click** on a sector: you then **toggle** the sector status.

A Purple sector will become Grey: That means that now the monsters cannot see the player anymore, and remains indiferent to the player.

A Grey sector will become Purple; That means that now monsters in this sector will see the player (provided no wall blocks their sight).

If you Shift and Click on the player's sector, it becomes Green. This sector is now safe, because monsters there cannot see the player, so cannot attack him.

Making sectors safe or unsafe

Select a sector, by clicking on it.

Then click on menu **Reject | Safe Sector**:
The selected sector appears in green, and the other sectors appear in grey.
Now monsters from any sector **cannot see the player,** when he is in the selected sector.

Then click on menu **Reject | Unsafe Sector**:
The selected sector appears in yellow, and the other sectors appear in purple.
Now monsters from all other sectors **can see the player**, provided there is a line of sight, when he is in the selected sector.

Saving the modifications

For security, the reject data is saved only when you decide it is correct.
Click on menu **Reject | Save changes**.

Things

WinTex cannot modify things. It cannot even list them.

Sorry. Use your favourite level editor for this!

Using custom textures in levels

Don't forget to read the tutorials.on texture editing.

Modify textures in a level

Once your level is displayed in the WinTex **Level editing window**, click on the button **WALLS** that you can find in the upper left panel.

Now, whenever you click on a line, on the level map, WinTex will display the textures associated to that line.

If you want to modify those textures, click on the menu **Texture | Change Textures**, and WinTex will display a panel that contains the list of available textures. The usage of this panel is rather self explanatory: just click on a texture in the list, and it will be put on the upper, lower or middle texture.

When you have finished modifying all the textures in your level, don't forget to click on menu **Texture | Save Change**.

Using a level editor to modify textures

WinTex is not as powerful as a level editor, for modifying textures. In particular, it doesn't do any automatic texture alignment. So you probably want to use your level editor for that task.

However, some level editors can use custom texture WADs, others just can't.

If your editor can use custom texture WAD, WinTex will attempt to tell to your editor the name of the WAD you are currently editing. This is done on the command line, see the configuration of the level editors.

If your editor can use custom texture WADs

Then from the WinTex main window, click on menu **File | Open WAD** and select your texture WAD, say, **TUTOR.WAD**.

You will have to load a level a level into this texture WAD.

Then you will start a level editor on that level, and WinTex will then tell your level editor that your **TUTOR.WAD** contains interesting information.

Normally, your level editor should read **TUTOR.WAD** at see that there are new textures there.

Then under your level editor, you should be able to use a new texture, called for instance **MYDOOR**.

If this doesn't work, check the configuration of your level editors. There are many tricks and bugs, and WinTex cannot compensate for them all.

If your editor cannot use WADs with custom textures.

Editor like DoomCAD, WadEd cannot (or could not, in previous versions) use custom textures.

So here is the WinTex trick, assuming that you are using **DOOM2** and that your texture WAD is called **TUTOR.WAD**.

> From the WinTex main window, in the panel **PWAD to join**, select the file **TUTOR.WAD**.
> Then click on the button **Merge in IWAD**.
> DeuSF will run for a while, and introduce **TUTOR.WAD** into **DOOM2.WAD**.

That way, any editor will be fooled, and beleive that **DOOM2.WAD** contains the textures listed in **TUTOR.WAD**.
You can start your editing session normally, and use a new texture, for instance **MYDOOR**, as if it had always been in DOOM2.
Create a level called **MAP01.WAD** and use **MYDOOR** as door texture.

When you have finished, click on button **Restore IWAD**, because you don't want **DOOM2.WAD** to remain modified forever, don't you?

Of course, this will mean that the level you have just created, let's say **MAP01.WAD**, will not run with DOOM2, because the texture **MYDOOR** doesn't exist in **DOOM2**.

Well, WinTex strikes again:

From the **WinTex main window**, click on menu **File | open WAD**, and select **TUTOR.WAD**.
The **WinTex Wad editing window** appears.

From the **WinTex Wad editing window** click on menu **Select | Level**, because you are going to load a new level into **TUTOR.WAD**.

Then click on menu **Edit | Load entry from file**.

WinTex asks for the name of an entry. Type **MAP01** for instance.
Or rather, select MAP01 form the drop box, that's faster.

WinTex then tells you that **MAP01** is replacing an entry from the main WAD.
This has better be! if it wasn't the case, your level name would be ignored by **DOOM2**.

Then WinTex ask you to point to a WAD file, **MAP01.WAD**.
This Wad should contain only one level. WinTex will only read the first level it finds, if the WAD contains more than one level.

WinTex then loads the level **MAP01.WAD** into **TUTOR.WAD**.
TUTOR.WAD is now complete, and ready to be used in the game!

Using custom Floors and Ceilings in levels

Don't forget to read the tutorials.on floor and ceiling editing.

Modify floor and ceiling textures in a level

Once your level is displayed in the WinTex **Level editing window**, click on the button **SECTOR** that you can find in the upper left panel.

Now, whenever you click on a sector, on the level map, WinTex will display the floor and ceiling textures associated to that sector.

If you want to modify those textures, click on the menu **Texture | Change Floors**, and WinTex will display a panel that contains the list of available floor and ceiling textures. The usage of this panel is rather self explanatory: just click on a texture in the list, and it will be put on the lower texture (the floor) or the upper texture (the ceiling).

When you have finished modifying all the textures in your level, don't forget to click on menu **Texture | Save Change**.

Using a level editor to modify floor and ceiling textures

Since most level editors cannot handle the new floor and ceilings correctly, it is not recommended to use your level editor to change floor and ceiling textures.

However, if you use a level editor on a WAD, from within WinTex, WinTex indicates to your level editor that this WAD contains new flats and textures.

Most level editors will just ignore this: then the only solution is to merge your WAD into the main WAD, so as to trick your level editor. Read the information about the use of textures, because the way to use your level editor with custom flats is very similar.

Testing WADs

Get an error report about your WAD

From the **WinTex main window**, click on menu **Examine | check PWAD**,
and select the WAD you want to check.

The tool **DeuSF** will run and generate an error report.
This error report will tell about possible **errors** in the **texture** definition, will signal textures that can't be used on **transparent** walls, and will tell about **undefined** textures in levels.
This check function is a bit **obsolete**. In future version it will be better, and integrated to WinTex.

Check if your WAD contains wasted space

From the **WinTex main window**, click on menu **Examine | List wasted space**,
and select the WAD you want to check.

If this is a WAD you are currently editing, you will notice there is a LOT of wasted space.
This is entirely **normal**: it's due to the fact that editing is reversible.

Of course, this will only **motivate** you to clean your WAD file before you distribute your WAD to other people (hey, don't forget to send it to me, by the way).

Tutorials of WinTex

These tutorials, written by various contributors, and checked by the author of WinTex, should help you quickly understand the basic tricks you need.

List of tutorials

Tutorial on Textures editing.
Tutorial on Floors and Ceilings editing.
Tutorial on Monster Sprites editing.
Tutorial on Pictures editing.
Tutorial on Level map editing.

Please **read those tutorials** before asking for support. Most of the questions you'd be tempted to ask are answered there, especially in the tutorial about sprites.

The most irritating question you'll ever ask

"I added new sprites in my WAD and they don't appear in the game"

This question was asked to me a **few hundred times** by e-mail and I'm real tired of it. So I answer once and for all: You need all the sprites in PWAD. Read the tutorial on sprites editing.

The newbie's corner

I have done large efforts to make WAD editing as easy as possible.
However, you must understand that when the guys at **id software** designed DOOM and decided how to organise the WAD files, the **last thing they considered** is how easy it would be for people to understand WAD origanisation, and to code editors.
They only bothered about their own code, and they arranged the WAD so that this code would not be too hard to write.

So there are many tricks and traps, that might turn a very ordinary desire into a nightmare.

If you're a newbie in WAD editing, be prepared to suffer some frustration at the begining.
I can only hope that WinTex will help you avoid most of the troubles.

Good luck! Have faith!
And don't forget to distribute your creations on the net!

Tutorial on texture editing

Creating and inserting a new wall texture is easy with WinTex, but don't be surprised if you don't understand it all on first try. This texture thing is real tricky if you're a newcomer in the DOOM editing world. However once you've learnt how to insert one new texture you'll find that it's just as easy to insert 50 or more...

For this tutorial, we assume that you are using **DOOM2**, and that WinTex is installed.
You will be editing a new PWAD file called **TUTOR.WAD**.

Menu

A bit of theory on textures.

Creating a new wall texture.

Using your own patches in wall textures.

Using custom textures in the levels.

The Advanced topics of texture editing.

Viewing the textures used in a level.
Open, with WinTex, the WAD this level is in. Then go to this level and click on the entry **SIDEDEFS**. WinTex will list all the textures used, together with the number of times they are used.

Tutorial on Texture: Theory

What is a Texture?

Countrary to the common belief, a wall texture is **not a picture**. It is only a **list of wall patches**, which are the actual graphic images. Sometime, this list is composed of only **one** wall patch, so you could confuse texture and patch.

A texture is composed by **placing wall patches** one after the other in specific position, with the later ones **sitting on top** of the earlier ones.

This 2-level approach can provide considerable **economy** of memory and disk space, since the same level patch can be used in more than one texture. A typical example is the **EXIT** patch, that appears in lots of different textures, at the exit of levels.

The different textures lists

The list of textures is usually stored in an entry called TEXTURE1.
This list is automatically created by WinTex if you create or modify any texture.

> Open **DOOM2.WAD** with WinTex, click on menu **Select | Textures**, and click on **TEXTURE1**, you will see that list of textures, with names and size of the textures.

Coming with the list of textures, you can find a list of patches, PNAMES.
You should not bother about this list, since WinTex handles it automatically, but you must know that it must be present, if you have added new patches.

> Now click on **PNAMES** and see the list of patches.

With DOOM and HERETIC, there is another list of textures, called TEXTURE2.
This list contains textures that are only in the registered version of the game.

> Open **DOOM.WAD** with WinTex, click on menu **Select | Textures**, and click on **TEXTURE2**, you will see that list of textures, with names and size of the textures.

Tutorial on Texture: creating a new texture

Start WinTex.
From the WinTex main window, click on menu **File | New Wad**.
Select **DOOM2** directory and give a name to your new Wad, for instance **TUTOR.WAD**.
The **WinTex Wad editing** window appears.

From the WinTex **Wad editing window,** click on menu **Advanced | Edit Texture**.
The **Texture editing window** appears.

Take a look at a texture

On the left down panel, you can see a **list of textures**.
Click on one texture, for instance **ASHWALL7**.
The Upper part of the screen is not displaying the texture ASHWALL7. This textures only takes a very small part of the screen, since WinTex is designed to edit textures of size 512x128.

Create a new texture

Click on menu **Textures | New Texture**.
You are requested to give the name of a texture.
Type the name **MYDOOR** in the drop box, and press **OK**.

Note that **MYDOOR** is inserted in the texture list, just above the last selected texture (so you will probably find it just above **ASHWALL7**).

Click on **MYDOOR** in the texture list.
A picture of size 128x128 appears, and it contains nothing.
Note that the color cyan means that the picture is completely **transparent**.

Bad news: if you give a transparent texture to DOOM2, it will crash, so you have to put something on your texture, and you'd better leave no transparent areas.

The things you can put on the textures are of course the wall patches.

Get the list of wall patches in main WAD

The right down panel contains the list of patches in your WAD. Since **TUTOR.WAD** is a new WAD, it contains no patches at all, the list contains patches from the **main WAD** file.

Put a wall patch on your textures

Click on a wall patch in the list of wall patches: for instance, click on **WALL00_1**.

When you click on **WALL00_1**, you notice that **WALL00_1** is displayed in the middle of the texture.

Now move your mouse above the texture: you will notice that **WALL00_1** follows the mouse.
Move **WALL00_1** to the left part of the texture and **click** on it, to drop it on the texture.

Note that **WALL00_1** was added to the list in the middle panel: This is the list of all patches for the texture **MYDOOR**, and for the moment it contains only **WALL00_1**.

Add more wall patches to your texture

Click again in the patch list, and drop more patches on your textures.

Drop patches until you **cannot see any cyan pixels**. That means your texture doesn't contain transparent areas, so it's usable without any problems.

You can also use textures with transparent areas (this is useful for windows) but due to restrictions in the DOOM engine, it's not as easy as it sounds.

Save your new texture

Click on menu **File | Quit**.
WinTex will ask you if you want to save the texture in the WAD. Answer **OK**.
The **Texture editing window** disapears (slowly, too slowly...).
The **Wad Editing window** appears.

In the **Wad Editing window**, you can see that your WAD now contains two entries, called **TEXTURE1** and **PNAMES**. This is normal.

Okay, now you've got a WAD with a new texture, and though it is probably not very artistic, you're eager to use it in a level.

Tutorial on Texture: using your own wall patches

Since now, you have used wall patches from the main WAD.
Now you certainly want to put your own pictures on the walls, and it's just as easy.

What you will do is simply: create a new patch in **TUTOR.WAD**, by using a .BMP picture.

Find some kind of picture, to use on the walls

Get your favourite Bitmap editor, and load any picture you think is real cool.
Reduce the size of this picture to a size of 100x100 (that's real small) and save it as **POSTER.BMP**.
It's going to become a great poster for your door.

See the general tutorial on Pictures to know how to create pictures suitable for DOOM2. You can use an ordinary picture, but it may look real lame when in **DOOM2**.

Create a new wall patch from that picture

We assume you are currently editing TUTOR.WAD. If you are not, then from the WinTex main menu, click on **File | Open Wad** and select the WAD where you want to put your new wall patch, for instance **TUTOR.WAD**. The WinTex **Wad editing window** will appear.

From the WinTex **Wad editing window**, click on menu **Select | Patches**.
This tells WinTex that you are now ready to edit wall patches.

Now click on menu **Edit | Read Entries from File**.
WinTex ask you for the name of an entry. This is the name that will be used in your WAD to represent the entry. Type **POSTER**, because you want a now poster for your door!
Click **OK**.

Now WinTex want you to select a .BMP file named, by default, **POSTER.BMP**.
Go to the right directory and click on the file **POSTER.BMP** that you prepared earlier.

WinTex should read that picture and put it into the WAD file.
An entry named **POSTER** should appear. Click on it, and you will see your picture.

Troubles with pictures?

Your picture could not be loaded?
Your picture's colors have changed and look lame?

See the general tutorial on Pictures to know more about the format of the .BMP picture that WinTex can accept.
WinTex is not a general purpose picture editor, it accepts only one (or two) formats.

Modifying your patch

Once you have inserted you patch in a WAD, you can modify it, like any other entry, by

selecting it and clicking on menu **Edit | Edit entry**.
Alternatively, just **double click** on the entry name.

This will save your entry to a temporary file and start a graphic editor on it.

Using your patch in texture.

Well, now that's real easy:

From the WinTex **Wad editing window**, click on menu **Advanced | Edit Textures** as usual.

The WinTex Texture editing window appears, as usual, and you can see the list of textures.

But now, you can **also** see your new patch **POSTER** in the list, on the lower right part of the window.

Click for instance on texture **MYDOOR**, in the list of textures.

then click on **POSTER** in the list of patches, and drop **POSTER** on the texture.

Here you are. A Door with a poster.

Tutorial on Texture: using custom textures in levels

Modify a level with WinTex

You need a level to try your new texture, so first, you will copy the level **MAP01** into your **TUTOR.WAD**.

From the **WinTex main window**, click on menu **File | open WAD**, and select **DOOM2.WAD**.
The **WinTex Wad editing window** appears.

Browse the list of entries, and go to the entry named **MAP01**, click on that entry and select all the following entries, until **MAP02** (not included).
You have selected all the entries in the level MAP01, now click on menu **Edit | Copy Entries**, to copy these entries into the clipboard. Then click on **File | Quit**.

From the **WinTex main window**, click on menu **File | open WAD**, and select **TUTOR.WAD**.
The **WinTex Wad editing window** appears.

Now click on menu **Edit | Paste Entries** and the level **MAP01** will be pasted from the clipboard into the file **DEFAULT.WAD**.

Modifying wall textures with WinTex

WinTex can let you modify textures in levels. This is somewhat rudimentary, though.

Now that **MAP01** is loaded in **TUTOR.WAD**, click on menu **Advanced | Edit Levels**, the WinTex **Level editing window** will appear.

Click on the level **MAP01** in the level list, on the lower left panel.
Click on button **WALLS**, on the upper left panel, just above the level list.

Now just click on any wall of the level. It will appear in green if it's a front line, or red if it's a back line. The upper, middle and lower textures on this line will be displayed below the level map.

Now you can click on the menu **Texture | Change Textures** and a list of textures will be displayed, from which you can pickup new names for the upper, middle or lower texture.

You can repeat the same operation for as many wall as you want. Once you have finished, though, click on menu **Texture | Save Changes** so that your changes are really taken into account.

Try your new level

Exit WinTex, and play TUTOR.WAD with:
doom2 -file tutor.wad

If it doesn't work, there may be a problem with your level editor, but read that tutorial again, just in case...

Tutorial on Texture:Advanced topics

Finding errors in textures

There is no texture check coded yet in WinTex, but you can use the one in DeuSF.
From the WinTex **main window**, select a WAD and click on menu Examine | Check PWAD.

DeuSF will run for a while, and try to report any obvious errors in the textures of the WAD.
It should warn you agains the medusa effects and column without patch errors.

Textures for transparent walls

You certainly want to create transparent textures for **walk through walls** or for **windows**.
You can, but there is a trick: If **any column** in your texture is covered by **more than one** patch, then something real strange, called **Medusa Effect**, will happen.
So **better** use **only one patch** on transparent textures, that is safer.

Errors due to columns without patches

It is a rule in DOOM that every texture column must be covered by at least one patch, even if that part of the patch is entirely transparent. Failure to respect this rule will cause a game crash with the message **column without patch**

Tutti frutti effect is also possible. This means that a transparent area is displayed, but there is nothing to display behind, so DOOM2 displays some random pixels

If you are composing a non-transparent texture the tutti-frutti effect can be caused by improper alignment of a background patch.   A simple example: a **128x128** texture composed of only one 128x128 patch whose upper left corner is pointed at the coordinate **1,0**.

A Wrap Around Sky

Despite what one could expect, the **sky** in DOOM and HERETIC **is a texture**, not a Floor or Ceiling.

But it's a special texture that <u>cannot be animated</u> and is displayed as if it had been mirrored horizontaly. When you will edit such a texture with WinTex, keep in mind that the real sky will be mirrored.

Replacing a sky texture requires re-using a DOOM2 sky texture name.
We'll reuse **SKY1**.

Compose four continuous patches **256x128** pixels each. Don't use cyan! but you can use some pale blue. Call those patches **SKYA, SKYB, SKYC, SKYD**.
Insert those four patches in TUTOR.WAD, by the usual method to insert patches.

Go to the WinTex texture editing window and select texture **SKY1**.
WinTex reports that this is the sky texture for **MAP01** to **MAP11**.

When you click on **SKY1**, you can see that the texture is of size 256x128.
Make it **1024x128** by clicking on the horizontal scroll.

Then select the patche **SKYA** and drop it at position **0,0**.
The position 0,0 will be indicated in a little text box above the texture.

Similarly drop **SKYB** at position **256,0**.

Now drop **SKYC** at position **512,0**.
You will need to use the horizontal scroll on the left, because WinTex can only display a 512x256 picture, so you need to move that picture a fair bit on the left.

Now drop **SKYD** at position **768,0**.

Here you are with your non-repeating warparound sky. Funny isn't it?
Try the levels MAP01 with TUTOR.WAD, go outside and see for yourself.

Note: if you're under Heretic, the procedure is the same, but the patches must be bitmaps of size **256x245**. However, the texture will still be declared as **1024x128**.

Animated Textures

New animated textures require re-using the Doom animated texture names, eg. **BFALL1**, **SFALL1**, **FIREBLU1**. Our example will use the **FIREBLU1** to **2** sequence (the 'cold fire' texture).

We will use original Doom patches: for the background **WALL40_1**, and for the animation the eight patches in the **VILEG1** to **VILEG8** sequence.

Go to the WinTex **texture editing** window.

There, select the patch **WALL40**.

Now in the texture list, select the texture **FIREBLU2**, the end of the **FIREBLU** animation.
You will notice that WinTex report them as start and end of an animation.

Create 6 new textures, called **FIREA**, **FIREB**, **FIREC**, **FIRED**, **FIREE**, **FIREF**.

Make sure that those textures are created between **FIREBLU1** and **FIREBLU2**, otherwise they won't be animated. That's why you must always select **FIREBLU2** before creating a new texture.

Now, for each of the texture **FIREBLU1**, **FIREA**, **FIREB**, **FIREC**, **FIRED**, **FIREE**, **FIREF**, **FIREBLU2**:
- replace the background patch by **WALL40**.
- Put one of the **VILEG1**, **VILEG2**, ... **VILEG8** patch.
Put the VILEGx pictures in sequence, and at roughly the same place, for better animation.

By the way: don't be surprised to use sprites on textures. This is well known since TRINITY.WAD by Steve McCrea, one of the best WADs ever.

Now you have the idea, which is to sandwich a sequence between an Animation Start and an Animation End, in that order. Note that it must be the same animation (FIREBLU here), and the Start must be placed before the End. Otherwise, you crash DOOM. Simple.

Go play any level that contains FIREBLU, and see the result. Poor Archvile is turning round and round!

For more detailed information about animated walls read the **Unoffical DOOM Specs** by **Matt Fell**.

Tutorial on Floors and Ceilings editing

This tutorial will help you understand how WinTex can be used to modify the appearance of floors and ceiling, a kind of entry that is called a **flat** by ID software.

For this tutorial, we assume that you are usinf DOOM2, and that WinTex is installed.
You will be editing a new PWAD file called **TUTOR.WAD**.

Adding new floors and ceilings to a WAD, and modifying them, is much easier than editing textures. WinTex now automatically handles all the tricks needed to have custom flats work in your PWAD file, so that you can concentrate only on the flat design.

Viewing the flats

First, you may wish to take a look at some flats, to know what it's all about.

From the WinTex **main window**, click on menu File | Open WAD. Select **DOOM2.WAD**.
The WinTex **Wad editing window** appears. There, click on menu **Select | Flats**
Click on any flat entry in the list of entry, this entry is displayed.

Adding a new flat to your wad

WinTex deals with flats (almost) the same way as with any other graphic entry.

From the WinTex **main window**, click on menu File | Open WAD.
Select **TUTOR.WAD**, the same name that is used in the Texture tutorial.

From the WinTex **Wad editing window**, click on menu **Select | Flats**.
Then click on menu **Edit | Read entry from file**.

WinTex will ask a name for the Wad entry, type any name you want, for instance **CARPET**.

Then WinTex will ask for a picture that will represent your flat.

WinTex will require a picture of size **64x64**. You should not use any other size for flats, or you may get real strange results. This size is a constraint of the DOOM engine, not a limitation of WinTex.

There must be no cyan pixels in your picture, or rather no transparent pixels. This is because flats cannot be transparent: there is nothing drawn behind.

Modify floor and ceilings with WinTex

WinTex can let you modify textures in levels.

Now that **MAP01** is loaded in **TUTOR.WAD**, click on menu **Advanced | Edit Levels**, the WinTex **Level editing window** will appear.

Click on the level **MAP01** in the level list, on the lower left panel.
Click on button **SECTOR**, on the upper left panel, just above the level list.

Now just click on any sector of the level. It will appear in yellow. The ceiling and floor textures of this sector will be displayed below the level map.

Now you can click on the menu **Texture | Change Floors** and a list of floors and ceiling textures will be displayed, from which you can pickup new names for the ceiling and floor texture.

You can repeat the same operation for as many sectors as you want. Once you have finished, though, click on menu **Texture | Save Changes** so that your changes are really taken into account.

Historical note

Old versions of the DOOM engine didn't allow to use custom flats in PWAD, so WinTex had a trick to turn this bug.

But, as was remarked by Alexander Prack and a few others, recent versions of the DOOM engine allow to use custom flats without any specific troubles. This method has be tested successfully on every recent version of the games.
So now it is implemented directly in WinTex.

The trick is simply to have a list of flats that doesn't begin with F_START (in WinTex, you will notice that it begins with FF_START, for security) and that does end with F_END.

Distributing a WAD file with modified flats

Thanks to the new method for handling flats, there is nothing specific to do before distributiong a PWAD with custom flats. In particular, installation of PWAD is not needed anymore.

Advanced Flat editing topics

Skip this if you deal with flats for the first time.

Don't replace the flat **F_SKY1**, it is never used by DOOM2.
It's a fake entry that is always replaced by the sky Texture, when the game runs.

As with other graphics, you should consult the Picture Tricks before you start serious work on a flat. The flats are especially sensitive to **Aliasing**.

When you insert a flat in a WAD, you will notice that WinTex automatically adds two entries called **FF_START** and **FF_END**. These entries indicate the start and end of the flat list. All the flats must remain together.

Tutorial on Installation batch

The tool DeuSF, combined with an installation batch, provides a secure way to deal with WADs that contain redefined Sprites.

This tutorial assumes that you are using DOOM2, that WinTex is installed, and that you have a WAD called **TUTOR.WAD** that contains custom sprites.

If you follow the instruction, you will be able to distribute a small WAD together with two other small files, and anyway it will install properly on any other PC.

Don't distribute WAD files that contain copyrighted data

First, you have to know that if a WAD contains redefined sprites, you can make it playable by appending all the sprites of **DOOM2**, or by merging it into the main WAD, **DOOM2.WAD**.

The idea is to avoid distributing a modified **TUTOR.WAD** or **DOOM2.WAD**, because it becomes real huge, and contains a lot of copyrighted data from **DOOM2**, so it's illegal to distribute it.

This is what the installation batch is for: it is a file that tells **DeuSF** what to do to install **TUTOR.WAD** on another PC, if **TUTOR.WAD** contains redefined sprites. It also create a **PLAY.BAT** file to play **DOOM2** with **TUTOR.WAD**, and a **RESTORE.BAT** file, so that **TUTOR.WAD** can be restored to it's initial state.

When other guys will want to play your WAD, they will just have to type **INSTALL**, and then when install is complete, they will type **PLAY** and that's it. It can hardly be made simpler.

Create an installation batch for your WAD

This is almost automatic.
From the WinTex main window, click on menu **File | Open WAD**.
Select **TUTOR.WAD**. The WinTex Wad editing window appears.

From the WinTex **Wad editing window**, click on menu **Advanced | Install WAD**.

The WinTex i**nstall batch window** will pop-up. This window is made of three panels.

You will notice **TUTOR.WAD** indicated on the top panel, and one check box for sprites: check this box if **TUTOR.WAD** contains redefined sprites.

The middle panel is dedicated to DeHacked patches. You can forget about it.

Go directly to the panel at the bottom and click on button **Create**.
WinTex will then quickly create the installation batch.

Press on button **View**, and a text editor will pop-up, that will display the **install.bat** file that was created.
Unless you're very good at **DOS**, this file will look fairly **cryptic**.
You don't need to understand that file.

Files that must be distributed with your WAD

Now you just pack:
- **deusf.exe**
- **install.bat**
- **tutor.wad**

in a single archive file, add a cool **tutor.txt** file that describes your new creation in enthousiatic terms, and spread it around! Don't forget to tell people that they must put all this in their doom2 directory, and type **INSTALL**, to run install.bat.

> if you are using PKZIP, just type
> **pkzip tutor.zip   deusf.exe tutor.wad install.bat   tutor.txt**

What is the advantage of using DeuSF?

DeuSF is the last version of a tool that I started developping in mid 94. It has been extensively used and tested, and is by far the most reliable (and the most powerful) tool to do the tricky job of sprite and flats installation. Other tools like NWT can do the job, though, but they are not as reliable.

"Thanx" to the new Borland C++ 4.5 libraries, DeuSF.exe has grown very fat, lately. But it still compressed to around 40K, which makes it smaller even than the tool NWT.

But what if the other guys don't have a PC with DOS?
Well, there are also versions of DeuSF for Linux, SUN O/S, OS/2, and even for strange machines like the Silicon Graphics and the unix DEC stations with processor Alpha. There is no DeuSF version for the Mac, though.

Tutorial on Level editing

WinTex is not a level editor. You will need the manual of a level editor for the more basic stuff.

For this tutorial, we assume that you are usinf DOOM2, and that WinTex is installed.
You will be editing a new PWAD file called **TUTOR.WAD**.

Load a level in a WAD with WinTex

We will explain here how to load a level in **TUTOR.WAD**.

From the WinTex **main window**, click on menu **File | Open WAD**, select **TUTOR.WAD**
From the WinTex **Wad editing window**,
Click on menu **Advanced | Edit Levels**.
That will start the WinTex **level editing window**.

This window is not a real level editor, you can just browse levels and do some very special reject tricks here.

Click on menu **Level | Load Level**.
WinTex will ask you for a level name, type **MAP01**.
WinTex will tell you that it's an existing level, which is rather normal since you are replacing MAP01, the first level.
Then WinTex will ask for a WAD file, that should contain at least one level. You could select DOOM2.WAD if you have no other WAD, but you could also select another WAD.

The first level in the WAD will be read, and loaded in TUTOR.WAD as level MAP01.

Now MAP01 must have appeared in the list of levels, on the left of the WinTex level editing window.

See level editing for more info on what you can see, and how to use your editor from withing WinTex.

Tutorial on Sprite editing

Derived from a text by **Graham Evans** (100101.3700@compuserve.com, author of Urban.wad and Moria.wad), with consent of author.

For this tutorial, we assume that you are usinf DOOM2, and that WinTex is installed.
You will be editing a new PWAD file called **TUTOR.WAD**.

Menu

1/ A bit of theory on sprites.

2/ How to browse the list of sprites.

3/ How to modify a simple.

4/ Adjust the offsets of sprites.

5/ Playing a WAD containing redefined sprites.

6/ How to modify the monsters sprites.

Preparations

From the WinTex main window, click on menu **File | New WAD**.
WinTex will ask for the name of a WAD, type **TUTOR.WAD**.

This will lauch the WinTex **main editing** panel, on a new WAD completely **empty**.

Now click on menu **Select | Sprites** to start a sprite editing session.

Tutorial on Sprite: Theory

There are three main types of sprites in Doom2: **Weapons**, **Monsters** and **Things**.

- Weapons are the pictures that represent the weapon you weild: chaingun, shot gun, etc.
- Monsters are the picture that represent the monsters, in various postures of movement, attack or agony.
- Things are objects like keys, lamps, bodies, etc..

We are going to change the barrels, a kind of thing.

By changeing a sprites, we do not alter how it behaves or what it is called, we just change **what a sprite looks like**.
(You can alter the way a sprite behaves by using **DeHacked**. Go to the WinTex editing panel, click on menu **Advanced | Install WAD**. You will find in the Install panel a small portion of the screen dedicated to DeHacked. Although I've done it, others are more qualified to explain it.

OK, so our new barrel is going to look different but will still explode if you shoot it or crush it.

Secondly, it is **NOT** possible to add completely **new** sprites to DOOM or DOOM2, you can only change existing sprites. (With dehacked you can change the name of sprites, but not create entirely new sprites. Well, not if you're new to this at least.)

All sprites for things in Doom are identified as follows:-

XXXXB0

Where **XXXX** is a four letter identifier, eg **PISG** for the Pistol, **BAR1** for the Barrel.
When you clicked on menu Select | Sprites, you may have noticed that **a drop box appeared**. If you take a look inside, you will see indications concerning the **conventional** names of sprites for DOOM2, and what those names are used for.
For instance you will see that **BAR1????** corresponds to barrels. This will be useful later, because the conventional names of sprites are not very intuitive nor very relevant.

Where **B** is a single letter for the sprite frame (the sequence)
The sprite frame starts at **A** and runs through in sequence, **B**, **C**, **D**, etc.
This is hard to figure out, but DeHacked will help you understand all this.
For monsters this can be different but not for this tutorial.

Where **0** indicates the point of view for the sprite.
A 0 as a number at the end means that it will look the same from all directions which is why the eye in the triangle always seems to move round with you. Monsters and certain things have different viewpoints for different views but we'll deal with that later.

Barrels have many parts. There are **2** pictures for the barrel sitting there and **5** pictures for the simple barrel exploding.

BAR1A0 **BAR1B0** for sitting there
BEXPA0 **BEXPB0** **BEXPC0** **BEXPD0** and **BEXPE0** for explosions

BAR1 means Barrel sitting there. **A-B** identifies the order of the frames (only two, to provide a very crude animation) and the 0 means it looks the same from all sides. (No problem 'cause it's round)

BEXP means Barrel exploding.
OK, now this bit you have to take my word for, **BEXPA0** and **BEXPB0** are the two frames you see when the Barrel is just about to explode.
BEXPC0, D0 & **E0** are the explosion.

Since these are just an explosion, by changing only **A0** and **B0**, we fool ourselves and others into thinking the whole explosion looks different.

Importing sprites from the main WAD

Theory over let's get to it.

Tutorial on Sprite: Browsing sprites

Go back to the WinTex main panel, and click on menu **File | Open WAD**. There, select **DOOM2.WAD** as the WAD to edit.
Don't worry, your precious main WAD won't be modified, even by mistake: WinTex has a protection to prevent modifications to the main WAD file.

This will start another WinTex editing panel, but this time on the main WAD file, so it's not really empty. There, click on menu **Select | Sprites**, to see only the sprite, otherwise you will be lost.

On the left is a box with all those funny looking names in it like **PISGA0** etc.
If you **click on a name** in this box, the window at right of the screen will show the picture that corresponds to this funny name. For instance, **PISGA0** will show a hand with a pistol.

Have a look at a few, weapons, monsters, things, they are all here.
You can click on the two buttons at the bottom of the list, to quickly jump from one sprite type to the next or previous one.

Page down the list about 4/5ths of the way and spot **BEXPA0 - E0**.   Click on each one, here are your 5 pictures of barrel explosion.

Tutorial on Sprite: Import a sprites

Now select all entries from **BEXPA0** to **BEXPE0** and click on menu Edit | Copy File. Nothing happens apparently.

Then go back to the panel dedicated to the new WAD, **tutor.wad**. For the moment this WAD is empty. There, click on menu **Edit | Paste Entries**. All of sudden the entries **BEXPA0 - E0** are transfered from the main WAD to the new **tutor.wad**!
Well, this will work only if the main WAD panel is still open. So don't close it.

Do the same for **BAR1A0** and **BAR1B0**.

Now the WAD tutor.wad contains the list of entries:

BEXPA0	10	28	=BEXPA0
BEXPB0	10	27	=BEXPB0
BAR1A0	10	28	=BAR1A0
BAR1B0	10	27	=BAR1B0

OK what has happened.
When you have requested to copy the entries from **doom.wad**, WinTex has stored the references of those entries in the clipboard (not the entries themselves).
When you pasted the entries, WinTex read those references and then copied the corresponding entries from **doom2.wad** to **tutor.wad**.
Then Wintex has updated the WAD directory, so that **tutor.wad** is now a valid WAD.

If you click on any of the entries, you will notice that the same picture than in **doom2.wad** will be displayed.
You will also notice, more to the right, that a small panel will appear, with strange indications: the x and y offsets of the pictures. Go to that panel, and click on the vertical and horizontal scrolls. See how they change the offsets, and how the picture moves. This is the easiest way to figure out what those strange offsets are used for!
More technically, the offsets determine the position of the sprite picture relatively to the point where the sprite is walking (this point is the center of the cross).

Editing the sprites

The simpler way to modify a sprite is to select an entry, for instance **BAR1A0**, and click on menu **Edit | Edit Entry**. Alternatively, just double-click on **BAR1A0**.

When you started editing your entry, the WinTex panel will temporarily disapear. This is only to prevent other modifications to the WAD. Don't worry, the WinTex editing panel will come back.

WinTex now will save a .BMP file to your temporary directory (C:\TEMP) and launch your favourite bitmap editor on that file.

Setup a bitmap editor

However, before you edit any bitmap entry, you must tell WinTex what editor you intend to use. WinTex will probably propose to use **PBRUSH.EXE** because it's the default Windows tool.
If you have a better tool, like **PSP 3.0**, refuse **PBRUSH**. Then WinTex will ask you to designate the editor of your choice, for instance **PSP30.EXE**.

Then WinTex will propose a command line for the tool. Just accept it, unless your bitmap editor is really particular.

Modifying the sprites

Under your bitmap editor, make some **modifications** to the picture of the sprite.
For instance, you could add a number on the barrel.

Be warned that the cyan color (light blue) that surrounds your barrel will be interepreted as **transparent**. So if you draw anything in cyan on the barrel, it will look like a hole has been poked into the barrel.

When you've finished, just **save** the bitmap file and **quit** your bitmap editor.
The WinTex editing panel, for tutor.wad, will then pop back.
Note that this panel will **not pop back** until you **have closed your editor**. So if you don't leave the editor, it will remain hidden forever.

Once the WinTex editing panel is back, WinTex will ask you if you really want to modify the entry. Click **OK**, unless the drawing you made didn't look good enough.

Then WinTex will ask you if you want to delete the temporary file. Click OK unless you intend to reuse that file for some other reason.

It is possible that WinTex complains about the **format** of the bitmap file. Please make sure that you saved your temporary file as a **256 color** Windows Bitmap, with no compression (**RGB**, not RLE. RLE is a kind of lame compression).

Tutorial on Sprite: Adjust the offsets

Now click again on your entry. You will notice that it has changed, and taken your modifications into account.

If you wish you can modify the offsets of that picture, and move it, for instance, more to the right, so that your barrel will look off-center: modify the horizontal offset by clicking on the horizontal scroll, in the offsets panel. You will notice that the X offset will increase, while your picture moves to the right.
Then press on **Apply**, in the offsets panel. This will write the offsets into the WAD file (Only do that when you are sure that the offsets are those you desire).

Now click on another entry, and click back on **BAR1A0**. The offsets have not been forgotten.

End of the sprite editing session

Just select menu **File | Quit**. Now tutor.wad is a valid WAD file, that can be used by other DOOM tools, or edited again by WinTex.

OK.   Next Beer or cigarette.

Tutorial on Sprite: Playing a WAD with sprites

Now, because you **cannot** just run a wad with a couple of new pictures in it (this is a DOOM bug) you MUST do the next steps in this order.

Go to the main WinTex screen.

then choose whichever solution suits you.

First solution: make a complete sprite WAD.

There, you will see a panel named **Wad to edit or modify**. There, select the WAD file **tutor.wad** that you have just finished to build. You may have to change the directory to **C:\tutor**.
Then press on the button **Add All Sprites** on the right of this panel.
DeuSF runs for a while, then a text editor screen will pop: this is only DeuSF telling you that it has modified your WAD **tutor.wad**. close this text.

You will notice tutor.wad has grown much **bigger**. It now includes **ALL** the sprites in DOOM2.

You can now play with: **doom2 -file tutor.wad**

the following is my start up line, remember my Doom2 is in **C:\Doom2**, my Wad is in C:\tutor directory, you will need to change this line to reflect your settings.

at the **C:\DOOM2>** prompt I type

```
doom2 -file C:\tutor\tutor.wad -nomonsters -warp02
```

This starts **doom2** using the wad I have created and goes direct to **map02** where I know there are some barrels and turns off the monsters so I don't get killed before I've got to those barrels!

Have a wander and look at your creation, then shoot one and watch it explode!

There you go, wasn't that simple.

Doom2 should not crash, despite you are using a modified sprite WAD.

When you have finished playing, select **tutor.wad** again and press button **Restore PWAD**. Press it only once! WinTex will then quickly tell you that your file tutor.wad has been restored, and actually it has dropped back to it's previous size and content. A sheer miracle!

Second solution: modify the main WAD

Note: **IWAD** is just another technical name for the Main WAD, **doom2.wad**.

The solution above is cool, because it doesn't modify the main WAD file. But it is very **slow**, and takes a lot of disk space.
Since the modifications that DeuSF makes are reversible, why not rather modify the main WAD file, and restore it later?

Go to the panel labeled **PWAD to Join**, and there, select **tutor.wad**. You may have to change the directory to **C:\tutor**.

Press the button **Merge in IWAD**, and let DeuSF run for a short while.

Your main WAD, **doom2.wad** has been modified.
In fact, **tutor.wad** has been merged to **doom2.wad**, and made into a bigger doom2.wad.

You can **browse** doom2.wad and see that the barrel is not the same as usual.

But rather, just play DOOM2 as usual: At the **C:\Doom2>** prompt type

```
doom2
```

You don't need to add -file tutor.wad, because tutor.wad is already in doom2.wad.

When you have finished playing, press on the button **Restore IWAD**, and doom2.wad drops back to it's previous state (well, that's a good thing, because you sure don't want to install those 15 megs again).
Note that if meanwhile you had delete **tutor.wad**, well tutor.wad is also restored when you restore the main WAD. Just some DeuSF magic.
The kind of touch that should convince you that DeuSF is really better than other tools ;-)

Distributing a WAD with modified sprites

If you are using new names for sprites, and using F_END, then you can directly distribute your WAD. This is the big advantage of using this tricky method. The disadvantage is that you need to modify the executable of DOOM2, with DeHacked, to change the name of some sprites.

Otherwise, you will need to use an installation batch using the tool **DeuSF**.

Tutorial on Sprite: editing monster sprites

Once you have figured out how to modify the barrel, modifying monsters is not very complicated.
The only additional difficulty is the viewpoints.

Go back to your doom2.wad editing panel.
Select the monster **TROO**, which is the Imp.
You will notice that **TROOA1** is the Imp seen from front, and **TROOA2A8** is the Imp seen from the right. Well, actually **TROOA2A8** is also the Imp seen from the left, but inverted first.

When you click on **TROOA1**, WinTex will display the picture, the offset panel, and a small panel under the offset panel. This panel is the **Viewpoints** indication.

For **TROOA1**, the viewpoint panels tells you that the **normal viewpoint** is Front (view) and the phase is A. Which is no more than an explicit decoding of A1.

For **TROOA2A8** , the viewpoint panels tells you that the **normal viewpoint** is Front Left (view) and the phase is A, but the **inverted viewpoint** is Front Right (view) and the phase is A.
This should be enough for you to understand that A2 codes for the normal viewpoint, and A8 codes for the inverted viewpoint. This means that when the sprite will be seen from Front Right, Doom2 will use the same picture as for Font Left, but mirrored horizontally.

So an entry which is related to a sprite has a strange name composed of five parts.
For instance let's take the sprite **TROOA2A8**:

Sprite name:	**TROO**	(means:an **Imp**)
Normal picture, phase:	**A**	(means: first **walking** phase)
Normal picture, viewpoint:	**2**	(means **front left**)
Mirrored picture, phase:	**A**	(means: first **walking** phase)
Mirrored picture, viewpoint:	**8**	(means: **front right**)

Apply this pattern strictly, when modifying the names of other sprite entries.
If WinTex cannot decode a sprite name, then something is wrong with it.

Tutorial on picture editing

For this tutorial, we assume that you are using **DOOM2**, and that WinTex is installed.
You will be editing a new PWAD file called **TUTOR.WAD**.

Viewing the pictures

First, you may wish to take a look at some pictures, to know what it's all about.

From the WinTex **main window**, click on menu File | Open WAD. Select **DOOM2.WAD**.
The WinTex **Wad editing window** appears. There, click on menu **Select | Graphics**
Click on any entry in the list of entry, this entry is displayed.

Adding a new graphic to your wad

From the WinTex **main window**, click on menu File | Open WAD.
Select **TUTOR.WAD**, the same name that is used in the Texture tutorial.

From the WinTex **Wad editing window**, click on menu **Select | Graphics**.
Then click on menu **Edit | Read entry from file**.

WinTex will ask a name for the Wad entry, type any name you want, for instance **TITLEPIC**.

Then WinTex will ask for a picture that will represent your graphic.

Select a **320x200** picture (that is, a picture of the same size of the original titlepic).

Modifying graphics from the game

This is not recomended, because those graphics are copyrighted.
However, if you really want to do it:
Keep **TUTOR.WAD** opened.
Form the WinTex main window, open **DOOM2.WAD** like you opened **TUTOR.WAD**.
Don't worry, you can't modify DOOM2.WAD by mistake. WinTex refuses to modify IWADs.
A new WinTex Wad editing window appears, that deals with DOOM2.WAD.
Click on menu **Select | Graphics**.
Now, in the list of entries, select entry **TITLEPIC** (or more entries).
Click on menu **Edit | Copy entry**.

Now go back to the WinTex Wad editing window than deals with **TUTOR.WAD**.
Click on menu **Edit | Paste entry**.
The entry **TITPLEPIC** from **DOOM2.WAD** is now in **TUTOR.WAD**.

You can close the Wad editing window that deals with **DOOM2**.

Warning: you can copy between files for **DOOM** and **DOOM2**, but not between files for **DOOM** and **HERETIC**. This is because pictures use a different color palette.

The picture tricks

Those are general tricks to use when creating Graphics for DOOM or HERETIC.

Those tricks are made necessary because of the limitations of the DOOM engine.
This engine is especially sensitive to aliasing, and unable to deal with colors correctly.

Reduction of picture colors (Color quantisation)

Most of the time, when you work on pictures meant for DOOM, you will work on RGB pictures (24 bit mode). But unfirtunately DOOM only supports 256 colors (8 bit color), and even worse: only the 256 colors of a special DOOM palette. It's the same for HERETIC, but the palette is different.
So you must fit the colors in your pictures down to that special palette.

Of course, the first step is to alway use colors that are close to the DOOM or HERETIC palette.

Here is how to fit your picture to the DOOM palette, using the cool Shareware PaintShop Pro 2.0.
- Load a reference picture file, extracted from the main WAD.
- Select menu **Colors | Save Palette** and save the palette to a file.
- Load the file you want to convert.
- Select Menu **Colors | Load Palette,** browse to find the palette, select it.
- In the 'apply palette to image using' box, possibly select Error diffusion. Select Ok.

Then save your file as **BMP - OS/2 or Windows Bitmap**, **Windows RGB Encoded**

DOOM will now display your Graphic pretty much as you expect. Experiment with this first, as your program might not do what you expect it to. For example ZSoft's PC Paintbrush will load the DOOM Graphic correctly and allow you to save the palette. However, if you take a 24 bit image and load the palette, nothing will happen. You can let PC Paintbrush convert the image to 256 colors (which it will do poorly), and then apply the palette. The results will be grotesque, because it doesn't quantize when applying a palette. A DOOM palette saved in this program would be useful only for Graphics you want to hand-draw from scratch. The moral is, if you don't know what you're doing, experiment first before using a Graphic you don't want to lose.

Note that WinTex supports **24-bit** bitmap (millions of colors), but color matching is slow as hell.
Avoid using Million colors with WinTex, always use 256 colors.

Smooth the pictures (Anti-aliasing)

The DOOM engine use a pervert technique when scaling down a picture (for instance, a monster seen from very far away): it suppresses lines and columns.
Do that, and at best, you end-up with a terrible looking pictures. At works, you are a victim of aliasing, the same phenomenom that happens when a sound is under-sampled. Hight frequencies and low frequencies mix up and the result can be painful to the eye (or the ear).

There is only one defense against aliasing: cutting the high frequencies, which, put in graphic terms, means smoothing your picture.
After you created a picture of a monster or a wall patch or a floor, you'd better apply one of those smooth or blur filters found in all graphic package. Be warned that they may only work with **24-bit** RGB picture, and thus you will need to quantise your color after smoothing.
Paint Shop Pro can do all that: convert your picture to **24-bit**, smooth it, load the DOOM palette and you've got a valid bitmap again, but much more suitable for DOOM this time.

Avoiding regular patterns (Anti aliasing)

You will want to avoid all repeating or regular patterns in your wall patches or your floors.
The reason is always the same: aliasing. A repeating pattern, especially if the pattern is short, corresponds to rather high frequencies, and when it's scaled down the resulting mess is terrible.
Good example of that mess are the yellow and black stripes in Aliens-TC, and the original floor of Alien-TC.
To avoid trouble, you will never use grids or meshes, unless they have a very irregular pattern. A grid of step 64 pixel will generate little aliasing, execpt at very far distance.
The rule is: the further your pattern shall be seen from, the least regular is shall be.