

Trespasser Menu Editing

By madppiper

What can I edit?

Well, the quick answer to that is: just about everything. You can for example add new images to the menu, edit the menu items, change the color of the progress bar (the one you see when you're loading a level)... well to make it short, you can modify the layout of anything that has nothing to do with the ingame rendering engine. And that is even quite simple to do.

So in order to change the menu, you'll need to open up the Menu folder in your Trespasser cd \data directory. (I assume you having copied the entire cd to your harddrive already). In there you'll find ddf, bmp and tga files. The bmp files and the tga files are images and the ddf files contain some code which tells the exe file what to display on the screen. Before we'll come to the editing part, I'll recommend you to make a backup-copy of all your files in that folder, since these files have some effect on the engine and might crash the game if you messed up at some stage...

if you're done with that, then let's bring it on!

What are DDF files?

The DDF Files in the menu folder of your trespasser cd /data directory contain several ddf files containing editable game data. Let me list you the files and give you an idea of what sort of data they contain:

audio.ddf

contains the layout of the audio-options menu.

controls.ddf

contains the layout of the controls menu.

directload.ddf

contains the layout of the quick level-load menu. (also known as the cheat menu)

gamewnd.ddf

displays messages /logos ingame... the developers deactivated two small images ("Internal use only ! Not for Editorial Review" and the EA logo)... which used to be displayed in-game....

hint.ddf

contains the layout of the hint option.

ingameopt.ddf

contains the layout of the main-ingame-options menu.

loader.ddf

contains the layout of loader, which is displayed when loading up a new level.

loadgame.ddf

contains the layout of the Loading menu.

mainscreen.ddf

contains the layout of the main menu.

newgame.ddf

contains the layout of the level-restarting-menu.

ngi.ddf

contains some general introduction. Possibly displayed in the demo or an early game presentation.

okdlg.ddf

contains the layout of an ok-button, which can be displayed ingame. (including some text).

options.ddf

contains the layout of the main-options menu.

options2.ddf

contains the layout of the ingame-options menu.

quit.ddf

contains the layout of the quit dialogue.

render.ddf

contains the layout of the video-options menu.

savegame.ddf

contains the layout of the savegame menu.

yesno.ddf

contains the layout of an yes-no dialogue.

But how can I edit these files?

Well, first of all you'll need to open them in a text editor, or another editor of your choice... once you got that done, you'll see a hole bunch of weird lines of code. Not let me explain that code on an example: mainscreen.ddf

Here's the code you'll see when you open up that file:

```
BACKGROUND 0 0 640 480 "menu\\ms_bkgnd.tga"
```

```
UICTRL "BUTTON" 1 1 1000 205 70 -1 -1
```

```
  1 "menu\\ms_new_0.tga"  
  "menu\\ms_new_1.tga"  
  "menu\\ms_new_2.tga"  
  ""
```

```
UICTRL "BUTTON" 1 1 1001 205 160 -1 -1
```

```
  1 "menu\\ms_load_0.tga"  
  "menu\\ms_load_1.tga"  
  "menu\\ms_load_2.tga"  
  ""
```

```
UICTRL "BUTTON" 1 1 1002 205 250 -1 -1
```

```
  1 "menu\\ms_options_0.tga"  
  "menu\\ms_options_1.tga"  
  "menu\\ms_options_2.tga"  
  ""
```

```
UICTRL "BUTTON" 1 1 1003 205 340 -1 -1
```

```
  1 "menu\\ms_quit_0.tga"  
  "menu\\ms_quit_1.tga"  
  "menu\\ms_quit_2.tga"  
  ""
```

```
UICTRL "BUTTON" 1 1 1006 550 160 -1 -1
```

```
  1 "menu\\trescom_logo.tga"  
  "menu\\trescom_logo2.tga"  
  "menu\\trescom_logo3.tga"  
  ""
```

```
UICTRL "BUTTON" 1 1 1004 0 430 100 479
```

```
  "" "" "" ""
```

```
;this is the static control for the AMD logo  
;just comment this control out to make it inactive
```

```
UICTRL "STATIC" 1 1 1005 550 393 -1 -1
```

```
  1 "menu\\sndelux.tga"
```

The first line 'Background' displays an background image. in this case it's the ms_bkgnd.tga file in the menu folder. The numbers before the "menu\\msbkgnd.tga" : 0 0 640 480, can be split down into two parts. 1) the position (in this case the image is displayed at the coordinates: x = 0 and y = 0. you can move the image around by changing one of these numbers.) 2) the image size (this however doesn't really seem to have much of an effect on how the image is being displayed).
The next five lines:

```
UICTRL "BUTTON" 1 1 1000 205 70 -1 -1
  1 "menu\\ms_new_0.tga"
  "menu\\ms_new_1.tga"
  "menu\\ms_new_2.tga"
  ""
```

place a button on a certain position in the mainscreen. Let's break these lines down as well. UICNTRL "Button" is the command that tells the engine that there is a button waiting to be displayed.

1 1 = is some data, which seems to be important to display that button, do not change any of these two numbers.

1000 = is the button id. (you have to give the button some sort of ID. if the id matches the one in the exe, then it'll execute the commands stored in the exe file. In this case it'll start a new game)

205 70 = are again the coordinates to display the button at.

-1 -1 =are in my beliefs some sort of closing tag, which close the first tag the *1 1*.

now let's check the next four lines in a row. The first line '*1 "menu\\ms_new_0.tga"*' displays the image of the button. The *1* tells the engine whether it's supposed to be displayed as a transparent graphic or not. (change it if you like and you'll see the original green surrounding of the image...).

The next line tells the engine which image it's supposed to display when you move the mouse over it and the line that follows , displays the image, when you click on the button.

The line after is like a closing tag for the entire button code. (never forget to add this to the code. Because if you leave it out, then the game won't display any other image unless it finds the next "" tag. If you leave it out in this case, then it won't display the following button: the load button)

Phew. That wasn't too difficult was it? Well take a look at the code again. You'll notice the lines stating:

```
UICTRL "BUTTON" 1 1 1004 0 430 100 479
  "" "" "" ""
```

that is nothing more than a blank button.

what follows are these two lines :

```
;this is the static control for the AMD logo
```

*;*just comment this control out to make it inactive is not relevant. it's just some comment that the developers inserted. It shows you however how to comment some lines. Simply add '*;*' before the line and the engine won't read and execute the lines following...

```
UICTRL "STATIC" 1 1 1005 550 393 -1 -1  
1 "menu\\sndelux.tga"
```

are the next lines to take a look at. This code is very similar to the code you're already aware of: the button code. The only difference is, that it displays a static image and that it therefore won't even need a closing tag (the "*;*")...

wow now you're capable of changing the first layout in Trespasser, the layout of the "main menu". 😊

But let's go on.

there are way more lines of code waiting for you to be found in the other files. Assuming that you've understood the basics of ddf file editing I'll simply show and explain the code to you...

```
UICTRL "checkbox" 1 1 1022 80 95 30 110  
"menu\\check_0.tga"  
"menu\\check_1.tga"  
"menu\\check_2.tga"  
"menu\\check_3.tga"
```

This displays a checkbox. Two displays are needed in case that the checkbox is enabled. the other two if it's inactive. (the second image is always the one being displayed on mouse_over).

```
UICTRL "textbox" 1 1 -1 120 93 300 110  
1 ; Transparent Bk  
1 ; Back Lit  
1 ; Back lit offset  
0 0 0; Back lit color  
"Play Voice Overs" ; Initial Text  
12 ; Point Size  
500 ; Weight  
0 255 0 ; Bk Color  
255 255 255 ; FG Color  
0x0 ; DT Flags  
0 ; Draw Border
```

This displays a textbox at a position of choice. There are quite a few options available:

Transparent BK = Transparent background(Yes=1 , no =0);

Back lit =gives the text some depth by adding a second layer of text underneath it

Back lit offset =set how close the second layer of text is being displayed to the original. In general this is set to 1, but I've noticed that it looks alot better if you make it 0

Back lit color = the color of the second text-layer

"Play voice overs"; the initial text

12 = Text-size

500 = Text-thickness (a higher number lets the text appear bold)

0 255 0 =Background color

255 255 255 = foreground, or Textcolor

0x0 ;DT Flags =format dword flags . meaning that it tells the engine how to align the text

0; Draw Borders = Draws a border around the Textbox, if not set to 0

Very similar to that are listboxes:

```
UICTRL "listbox" 1 1 1002 10 10 300 180
  12 ; Point Size
  500; Font Weight
  1 ; Allow Selection
  1 ; Transparent Bkgnd
  0 255 0 ; Transparent Bkgnd Color
  1 ; BackLit Text
  1 ; BackLit Offset
  0 0 0; BackLit Color
  128 128 128 ; Background color
  255 255 255 ; Foreground color
  1 ; Show VScroll Always
  0 ; Display Border
```

it has the same options, but differs by allowing us to select something. That's allowed by the "1 ; Allow Selection" code.

Well and to round this up, we've also got the editboxes:

```
UICTRL "editbox" 1 0 102 0 0 300 20
  14 ; Point Size
  500 ; Weight
  1 ; Transparent Bk
  1 ; Back Lit
  1 ; Back lit offset
  0 0 0; Back lit color
  "" ; Initial Text
  0 255 0 ; Bk Color
  255 255 255 ; FG Color
  0x24 ; DT Flags
  0 ; Draw Border
```

These allow users to enter some text. The initial text is then overwritten...

Next in line are hotspots.

```
UICTRL "hotspot" 1 1 1102 80 95 300 110 0
```

Hotspots allow users to click on a certain area and then function very similar to a button.

And that's not nearly it. You can even change the progress bars you see when you load up a level. (the loader.ddf file contains the data for that). The code looks like the following:

```
UICTRL "progress" 1 0 100 29 81 288 84
  0; Frame Rect
  0; Min Range
  99; Max Range
  0; Curr
  1; Step
  255 65 0 ; FG Color
```

0; Frame Rect = thickness of the progress bar

0; Min Range = Minimum Range of the progress bar (in this case 0)

99; Max Range = Maximum Range of the progress bar (in this case 99; In combination with the Min Range you can see that this progress bar ranges from 0% to 99%... You can set it even higher if you prefer a more detailed and longer Progress bar)

0; Current = Starting point of the progress bar

1; Step = Each time the engine passes one of the loading cycle, it'll add 1 dot on the line... if you increase this amount, then it'll reach the end of the line faster... doesn't mean however that it'll load faster...

255 65 0 ; *FG Color* = Foreground color or color of the progress bar...

The only thing left to explain is the nice sliders, which can be seen in the audio menu, or the video options menu...

```
UICTRL "slider" 1 1 1006 39 83 185 97
    "menu\\slider.tga"
    5 ; # of units
    0 ; Default Unit value
    0 ; Border/Ticks
```

"menu/slider.tga" = displays the slider.tga image as the slider bar...

5 ; # of units = tells the computer how many steps there are for you to select... the higher, the more fluent you'll be able to increase an option

0 ; Default Unit Value = Default setting for the slider

0 ; Border / ticks = display a slider-bar underneath the slider... not all too bad looking if you ask me.... (you can turn it on by changing the 0 into a 1)

Well, we're done with the heavy part 😊 Wasn't difficult, was it? Hope you'll enjoy playing around with it. With this it's actually quite simple to create your own theme for Trespasser...

How can I change images?

This is easy. You can either edit the existing ones or create entirely new ones. If you want to create new ones, then you will have to follow two rules:

- 1) You can only use Bitmap (.bmp) or Targa (.tga) files. Others will crash the game...
- 2) The images must use an RGB color setting!
- 3) If you want to change an animation (eg the ones on a button) then remember to use the same filetype. Otherwise you'll see the image twice... 😊 (since the old image won't be replaced...)

That's it 😊 Happy editing...

How can I change the mouse-arrow?

Simply open up the mouse.bmp file and change it a little. you can even insert a mouse button from a different game... all of it should work 😊