Doomsday Engine - Feature #1580

Fix the Doom 64: Absolution TC plugin

2013-05-10 10:41 - dekuius

Status:	New	**Start date:**	2015-04-30
Priority:	High	**% Done:**	30%
Assignee:			
Category:	3rd party compatibility		
Target version:	Modding		

Description

Subtasks:

Bug # 2021: Doom64's ingame music is severely broken	**New**
Bug # 2027: Doom64TC map is missing 'on' textures for switches	**New**

Related issues:

Related to Bug #2019: ST_UILogForPlayer returning garbage in jDoom64	**Closed**	**2015-04-30**
Related to Feature #2023: New Doom 64 plugin (accurate gameplay, original ROM)	**New**	**2015-05-02**
Related to Bug #2022: p_enemy.c in doom64 plugin uses ~15 different copies of...	**Closed**	**2015-05-02**
Related to Feature #2029: Get the Doom64 lighting system working in Doom64TC	**Rejected**	**2015-05-03**

History

#1 - 2013-10-18 20:34 - skyjake

- Tags set to Doom64

- Subject changed from doom 64 support to Doom 64 support

- Category set to 3rd party compatibility

The doom64 plugin has been kept around and compiles fine, but likely needs plenty of fixing.

#2 - 2013-10-18 21:37 - danij

This plugin is also far from feature complete. It was built by extracting the DOOM64 specific logic from the Absolution TC project which was itself built on a much older version of Doomsday (1.7.8 / 1.7.14 as I recall). The game logic depends on data which has been "manually" extracted from the DOOM64 IWAD embedded in the ROM.

DOOM64 support in Doomsday should not have to rely on this preprocessed IWAD and should instead work given only a path to a ROM file. In other words, Doomsday should treat the ROM as an IWAD and extract and convert the data itself.

Also, much of the game logic in this plugin is based on guesswork and approximation. It should be replaced using the logic of DOOM64 Ex as a guide.

#3 - 2015-04-30 00:31 - rhargrave

- File libjd64-running.png added

Some notes about the plugin, for those that are interested:

I have forcibly made the plugin to accept the 33-Map (rather than 38-Map) IWAD, as even generating a WAD from a ROM.

That aside, here's what is working.

- Menu music seems fine

- Font works

- Skull icon for selected item works, is animated

- Menu logo textures work

What's not behaving:

- The engine skips *alot* of patch files, complaining that the corresponding maps do not exist

- FPV Weapon sits visibly above the bottom of the screen. This is only noticeable with a non-opaque status bar, and is also present under the same conditions when D64 resources are loaded in to doom2 and played (not that you should do this, many things will not work, and are not meant to work).

- Missing lumps

- Slider textures are borked

- Save game slots appear to be missing backgrounds

- There are colour inconsistencies between menu prompts and menu items

- Reloading the game results in GUI_ShutDown calling `free` on an invalid pointer

- In-game music is horrifyingly not music. If testing, it is *highly* advisable, that, unless working on music, you mute the ingame music.

- Logging to the OSD appears to cause a segmentation fault. For example, picking up an item results in this call chain: `P_TouchSpecialMobj (libdoom64) > P_SetMessage (libdoom64) > UILog_Post (libdoom64) > Z_Realloc (in libdeng_legacy) > [ memcpy implementation ]`. For me, `memcpy implementation` is `__memcpy_sse2_unaligned`. However, I am unsure whether this is SSE-specific or not -- I will look in to this. As it stands, this is the most breaking issue I have yet to encounter.

Code Things:

- A method borrowed from jHexen is used when firing the lazer weapon (this is indicated in the source code)

- There are a *lot* of magic numbers/ambiguous values in the plugin code marked only as 'jd64' or '64tc' without any explanation as to the purpose

- While the plugin is far from feature complete, it shares a very large amount of code with the jDoom plugin. it should probably be investigated whether or not some kind of code sharing could be done, if possible. I do not, however, think that jDoom64 should merge in to jDoom as there are too many significant modifications to make that possible.

- Some C files need conversion to CPP, and need some logic cleanup and auditing, also.

I have attached a screenshot at the main menu with a portion of the warnings shown, not for debugging purposes, but simply for anyone interested in seeing the TC WAD loaded.

#4 - 2015-04-30 08:06 - skyjake

- Related to Bug #2019: ST_UILogForPlayer returning garbage in jDoom64 added

#5 - 2015-04-30 13:53 - skyjake

Good list. This should be a good basis for creating some subtasks under this issue.

#6 - 2015-05-02 15:46 - skyjake

- Related to Feature #2023: New Doom 64 plugin (accurate gameplay, original ROM) added

#7 - 2015-05-02 15:52 - skyjake

- Tags changed from Doom64 to Doom64, Doom64TC

- Subject changed from Doom 64 support to Fix the Doom 64: Absolution TC plugin

I think it is good to be clear that the current Doom 64 plugin is completely based on the Doom 64 TC, so this issue should be about repairing and updating the code to work with modern Doomsday. The end result is an up-to-date version of the Doom 64: Absolution plugin.

There is a separate line of work regarding the creation of a new, more accurate Doom 64 plugin (#2023), although that work is not yet even started.

I think there is a case to be made for having a plugin for both the Doom 64: Absolution TC and the "real" Doom 64 emulation. The former is a matter of fixing what already exists, while the latter is a major effort involving writing new code and taking care of possible Boom/MBF dependencies.

#8 - 2015-05-02 15:53 - skyjake

- Related to Bug #2022: p_enemy.c in doom64 plugin uses ~15 different copies of A_KeenDie added

#9 - 2015-05-03 14:38 - skyjake

- Related to Feature #2029: Get the Doom64 lighting system working in Doom64TC added

#10 - 2016-08-09 11:04 - skyjake

- % Done changed from 0 to 30

#11 - 2019-11-29 12:31 - skyjake

- Target version set to Modding

#12 - 2019-11-29 21:21 - skyjake

- Related to Bug #2027: Doom64TC map is missing 'on' textures for switches added

Files

libjd64-running.png	1.21 MB	2015-04-30	rhargrave