Doomsday Engine - Bug #2436

Incompatibility with the new M1

2020-11-24 12:51 - applecrypt

Status:	In Progress	**Start date:**	2020-11-24
Priority:	Normal	**% Done:**	100%
Assignee:	skyjake		
Category:	3rd party compatibility		
Target version:	2.3.1		

Description

Just upgraded to the new Mac Mini (M1) 8GB/512GB HDD.
Re-downloaded Doomsday (both for stable and then unstable) but neither works properly. The game loads either version of Doom okay - but the main menu text is blurred out red. I can still use this menu so started a new game. The textures aren't working and everything looks quite plain and cartoon-like.

I'm using the same WADs for Doom1/2 that I've been using for years with Doomsday. So, I'm assuming this is something to do with the new M1 processors?

Fwiw, Doom 2 says "Missing Data Files". But, these are all the same WADs I've used for years :?

Hoping this helps?

Cheers
Steve

Associated revisions

Revision e99ebb16 - 2021-01-08 12:50 - skyjake

macOS|Fixed: Missing textures on M1 Macs

For some reasons, using a switch-case when sampling the batch textures fails to select anything above batchIndex 0 on the M1 Macs. A likely shader compiler bug.

IssueID #2436

Revision 482c83e2 - 2021-01-08 13:42 - skyjake

macOS|Fixed: Invalid map culling

In an arm64 build, conversion from float/double to uint16_t behaves differently than in x86_64: half of 16-bit binangles were zeroes. Convert first to int32_t.

IssueID #2436

Revision a79bf1bb - 2021-01-08 21:38 - skyjake

macOS|Fixed: Missing textures on M1 Macs

For some reasons, using a switch-case when sampling the batch textures fails to select anything above batchIndex 0 on the M1 Macs. A likely shader compiler bug.

IssueID #2436

Revision 722e3416 - 2021-01-08 21:55 - skyjake

macOS|Fixed: Invalid map culling

In an arm64 build, conversion from float/double to uint16_t behaves differently than in x86_64: half of 16-bit binangles were zeroes. Convert first to int32_t.

IssueID #2436

Revision a564c311 - 2021-02-03 13:32 - skyjake

Fixed|macOS: Bullet angles restricted to 0-180 degrees

IssueID #2436

Revision 5a4b184e - 2021-02-03 13:33 - skyjake

Fixed|macOS: Bullet angles restricted to 0-180 degrees

IssueID #2436

Revision 52c921f0 - 2021-02-03 13:39 - skyjake

Fixed|macOS: Camera doesn't move downwards

Another float-to-unsigned cast that clamps negative values to zero.

IssueID #2436

Revision 78076b29 - 2021-02-03 13:41 - skyjake

Fixed|macOS: Camera doesn't move downwards

Another float-to-unsigned cast that clamps negative values to zero.

IssueID #2436

History

#1 - 2020-11-24 13:18 - skyjake

Sounds like Rosetta 2 is running the CPU code fine but the OpenGL rendering side is having some hiccups. The M1 has Apple's own GPU so it is running different OpenGL drivers than before, and with the focus on the Metal API it seems unlikely these drivers will improve much in the future.

I don't have an M1 Mac myself so I'm unable to find workarounds for these rendering glitches for the time being.

In the slightly longer term, switching to Metal for the rendering would be advantageous on the Mac.

#2 - 2020-11-24 14:16 - applecrypt

> In the slightly longer term, switching to Metal for the rendering would be advantageous on the Mac.

I hope this is something that isn't years away? I'm a bit gutted to lose Half-Life 2 and then I find out Doomsday hiccups with the new M1... Not a good day! Thanks for the reply :)

#3 - 2020-11-25 11:08 - skyjake

I feel you... Being an early adopter does have it downsides. :/

Hmm, a full Metal port is a significant undertaking, but perhaps there is an intermediate step that could be taken. I'm much closer to getting an OpenGL ES 3 version up and running, suitable for Android and Raspberry Pi, for example. GLES3 might work better on the M1 since it is quite a bit simplified compared to the full OpenGL. And if not, there's always solutions like ANGLE that will help.

#4 - 2020-11-25 11:09 - skyjake

- Category set to 3rd party compatibility

- Target version set to Rendering

#5 - 2021-01-08 12:53 - skyjake

- Subject changed from Incompatibility with the new M1 ? to Incompatibility with the new M1

- Status changed from New to In Progress

- Assignee changed from Deng Team to skyjake

- Target version changed from Rendering to 2.3.x

- % Done changed from 0 to 50

Good news, I now have the pleasure of owning an M1 MacBook Air and have found a workaround for the missing texture issue.

It's a bit bizarre and looks like a bug in Apple's shader compiler implementation. Changing from a switch-case to a series of if statements fixed the problem.

There is another rendering problem with the geometry culling, though. Likely a problem with integer overflow behavior, just need to find out exactly where it happens.

#6 - 2021-01-08 13:43 - skyjake

- % Done changed from 50 to 100

#7 - 2021-01-08 13:44 - skyjake

- Target version changed from 2.3.x to 2.3.1

#8 - 2021-01-08 14:33 - applecrypt

Brilliant!!! And congrats on a fine purchase too. I'll look at downloading the latest version later - I'm assuming it'll be the nightly build?

#9 - 2021-01-08 14:40 - skyjake

applecrypt wrote:

> I'll look at downloading the latest version later - I'm assuming it'll be the nightly build?

The fix will be heading to 2.3.1 first. Due to the 3.0 transition in the master branch I can't release an unstable build right away, but I'll bump the stable branch to 2.3.1 RC and include it there. It should be available (approximately) by tomorrow for testing.

#10 - 2021-01-08 16:21 - applecrypt

Fair enough, I don't understand the techno-jargon behind this ;) but I'll look at the RC version update tomorrow/the weekend. Should be fun! I will give it a playtest :)

Files

Screenshot 2020-11-24 at 10.49.17.png	72.5 KB	2020-11-24	applecrypt
Screenshot 2020-11-24 at 10.49.35.png	154 KB	2020-11-24	applecrypt