

Emulator - Emulator Issues #5682

Zelda SS Crash when entering a silent realm trial (in newer revisions only)

11/01/2012 12:21 PM - scientificraver

Status: Fixed	% Done: 0%
Priority: High	
Assignee:	
Category:	
Target version:	
Operating system: Windows	Relates to performance: No
Issue type: Bug	Easy: No
Milestone:	Relates to maintainability: No
Regression: No	Regression start:
Relates to usability: No	Fixed in:

Description

In recent dolphin revisions when entering one of the 3 trials in Zelda SS, dolphin crashes. The crash happens in all current revisions but in 3.0-631 and older everything works fine, although I did not have the time yet to track down the exact revision where the error occurred first.

1) Game Name and ID (as it appears in right click > properties:
SOUP01

2) What is the expected output? What do you see instead?
Dolphin not crashing is expected

3) Did the game ever work correctly (i.e. not have this problem) on an earlier version of dolphin? Please specify the exact revision when the problem began.
in 3.0-424 and 3.0-631 it worked fine all the time

4) What steps will reproduce the problem?
1. Enter one of the silent realm trials
2. After the cutscene where Link puts his sword into the stone dolphin crashes

5) Win7x64, mobile CoreI7 GT555M, 3.0-799 HLE, 1080p 3xIR

Related issues:

Has duplicate Emulator - Emulator Issues #5883: Skyward Sword - Crash with DX11	Duplicate
Has duplicate Emulator - Emulator Issues #6281: Skyward Sword and Mario Galax...	Duplicate

History

#1 - 11/04/2012 07:50 AM - Aaron5367

I also experience this issue in the latest releases. Had to fall back to 631, as said above, and it worked fine then.

1) Game Name and ID (as it appears in right click > properties:
SOUE01
Zelda: Skyward Sword

2) What is the expected output? What do you see instead?
Dolphin not crashing is expected

3) Did the game ever work correctly (i.e. not have this problem) on an earlier version of dolphin? Please specify the exact revision when the problem began.

All the latest releases on the main page (except 3.0, which I wasn't able to check) do this. 3.0-631 works fine.

4) What steps will reproduce the problem?

1. Enter one of the silent realm trials
2. After the cutscene where Link puts his sword into the stone dolphin crashes

5) Windows 7 x64, AMD Phenom II X6 3.6 GHz, 10 GiB Ram, AMD Radeon 6970, 1080p Auto IR, HLE

#2 - 11/04/2012 11:22 AM - scientificraver

The best we could do to support the developers might be to track down the exact revision where the error starts to occur. If 631 works and 799 not, we would need to test something like 650, 700, 750 and 775 next, to get closer to the relevant changes in the dolphin code.

#3 - 11/16/2012 03:43 AM - limburgerite

I just tried OpenGL on a few 'broken' builds and it seems to be unaffected (though it did crash on me once out of about eight tries). DX9 and DX11 crash every time, though.

I've narrowed it down to builds 772 and 774 - 772 works, 774 doesn't. 773 isn't in the download archive, but this is the rev between 772 and 774: <http://code.google.com/p/dolphin-emu/source/detail?r=bb8b5936c040bcf693b270d65821185386ff83b1>

I don't think 774 actually broke it since it was only a minor DX11 change, and the crashes occur with DX9 as well.

#4 - 11/16/2012 03:48 AM - skidau

Try clearing the shader cache folder.

#5 - 11/16/2012 04:38 AM - limburgerite

While deleting the shader cache has cleared up issues for me before, all the tests I've done for this bug have been with freshly extracted downloaded zips. The only changes I've made after extracting are copying my Wii folder (for the SS saves), setting the fullscreen res to 1920x1200 (I get a black screen with 640x480), and setting Real Wiimote.

#6 - 11/19/2012 12:13 PM - NeoBrainX

Thanks for tracking down the issue. Still, could you please verify that the issue is indeed caused by rbb8b5936c040? Frankly said, I still find it very unlikely to be the case.

#7 - 11/19/2012 12:15 PM - NeoBrainX

If possible, can you create a fifo log (with an older build where this issue doesn't occur) of the frame were the crash occurs in newer builds? It doesn't need to be a single-frame log, a log of multiple frames including that specific one would be sufficient.

#8 - 11/21/2012 05:58 AM - limburgerite

I think I may have come a bit closer to identifying the problem.

Long story short, the shader cache files generated by build 772 work in 774 (and 840 and 843 and probably all post-772 builds), both DX9 and DX11. I simply went past the crash point in 772, took the cache files and copied them to the broken builds. Presto! No crash. There's another cutscene shortly after entering a realm (when you exit the circle) that also causes a crash, but if you go past that point with 772 it can be avoided as well.

So it seems to be a pixel shader generation/compilation issue. Some evidence to support that theory:

* rbb8b5936c040 is the only revision between between working build 772 (r8fed3b76c85e) and broken build 774 (rac2ce8b16ee5). rac2ce8b16ee5 was only a minor modification to the DX11 plugin, but rbb8b5936c040 modified PixelShaderGen.cpp.

* It's necessary and sufficient to copy the good ps cache files to stop broken builds from crashing (the vs files have no effect).

* Builds before 135 (radef86c1ef29) are actually broken in the same way as recent builds. It seems the 'new-shadercache-uids' merge fixed this issue. Could rbb8b5936c040 have reverted PixelShaderGen.cpp to a pre-merge state? I'm still trying to wrap my head around the reversion of a reversion thing...

* I let VS2010 debug the crash a few times and it's crashing in D3DCompiler_43.dll.

For the record, these are the builds I've tested so far:

working: 135, 139, 153, 682, 714, 721, 735, 759, 765-772

broken: r7231, r7719, 77, 97, 774, 776, 793, 804, 831, 840, 843

Oh, and here's the FIFO file you wanted, Neo:

<http://www51.zippyshare.com/v/8903973/file.html>

Sorry for the large-ish size; it's 75 frames. The crash always occurs in the solid white between scenes so it's hard to pin down the exact frame. All settings were at default except 'Real Wiimote'.

Some info from the VS2010 debugging:

<http://www44.zippyshare.com/v/20010327/file.html>

I tried setting the next statement and continuing and it actually popped up the 'failed to compile shader' dialog and dumped a bad_ps_0000.txt. A couple are included in the above file.

#9 - 12/01/2012 03:41 PM - NeoBrainX

Uhm, not sure if I ever asked someone to do this - but what happens if you set the EnableShaderDebugging field in your gfx config to true? The game should slow down quite a bit, but hopefully it also gives a more useful error dialog when crashing (or maybe not crash at all but still show an error message). Make sure to have panic handlers enabled when testing.

#10 - 12/01/2012 09:01 PM - limburgerite

OK, I set EnableShaderDebugging = True in gfx_dx9.ini (Windows x64 build 863 non-dirty) - it slowed down to about 10-15 fps, so I'm assuming it worked. It crashed in the same place; however, after I did the debug/set-next-statement/continue, I got some more dialogs after the 'failed to compile shader' one:

Warning

Unique pixel shader ID mismatch!

Report this to the devs, along with the contents of ./User/Dump/psuid_mismatch_0000.txt.

OK

Nine of them in a row, then it crashed again in the same spot.

the psuid_mismatch_000?.txt files:

<http://www24.zippyshare.com/v/55676305/file.html>

(Can I just attach this one? It's like 3KB)

#11 - 12/02/2012 03:03 AM - NeoBrainX

Thanks, that should help debugging this issue a lot.

Attaching 3 kb files is fine, we just don't want people to upload their 1 MB-sized BMP files of screenshots etc. ;)

#12 - 12/02/2012 11:57 AM - NeoBrainX

Yo, the application behavior when continuing program execution after a crash usually is undefined, so that information is not really reliable ;)

Do you have a backtrace of the crash?

#13 - 12/02/2012 05:31 PM - limburgerite

As luck would have it, I found an old Dolphin git repo I had zipped up but never got around to doing anything with. So I finally went ahead and installed the DX SDK and managed to build a debug exe.

It's r62e790f, which is over a year old, but it does have the issue. Sure enough, it crashed in the exact same place.

I had to add `#include "ConfigManager.h"` to `EmuWindow.cpp` to get it to compile, plus make a fake `scmrev.h` (I don't have git extensions installed yet), so the build's a tad dirty, but other than that the source is untouched from r62e790f.

I've attached the crash debug info (no continue this time, just breaking at the crash)

#14 - 12/02/2012 05:48 PM - NeoBrainX

Could you try replacing this (<http://code.google.com/p/dolphin-emu/source/browse/Source/Core/VideoCommon/Src/PixelShaderGen.cpp#487>) line with `"static char text[16383];"` (i.e. decrement the value by 1) ?

#15 - 12/02/2012 05:52 PM - NeoBrainX

That will either change nothing or make an error box appear before crashing. The former is much more likely, but I really have no idea what would be causing this.

#16 - 12/02/2012 06:08 PM - limburgerite

I just added some code to `CompilePixelShader()` that logs the shader code passed to it. It spit out `shader_%.d.txt` files from 1-20 before the crash. I tried compiling some of them with `fxc.exe` from the SDK. They all produce output except `shader_20.txt` - which actually crashes `fxc.exe`. Uhm, is that supposed to happen? If I screw around with a good shader file, it gives errors, like you would expect.

I'll try your suggested mods next.

#17 - 12/02/2012 06:10 PM - NeoBrainX

I guess it's not supposed to crash `fxc.exe`... I'm starting to think this is an actual shader compiler bug :/

Could you attach that `shader_20.txt` shader and maybe the `shader_19.txt` one, too?

#18 - 12/02/2012 07:22 PM - limburgerite

(Sorry for the delay - had a little F7U12 moment after a hard lockup)

The 16384->16383 change didn't have any effect.

I changed my code to "shader_%03d.txt" (like it should've been in the first place) and ran it again, so now the crasher is shader_016.txt (still the same file as the previous shader_20.txt though)

#19 - 12/02/2012 08:51 PM - limburgerite

Hmm... Disabling optimizations lets fxc compile the 'bad' shader.

Looking more and more like a M\$ bug... Though I still find it interesting that the 'new-shadercache-uids' merge fixed the issue.

I'll probably have to bow out after this since my knowledge of shader programming falls somewhere between jack and squat.

#20 - 12/02/2012 08:56 PM - NeoBrainX

You could try replacing lines like this

```
"crastemp = frac(rastemp * (255.0f/256.0f)) * (256.0f/255.0f);"
```

with

```
"crastemp = rastemp".
```

Maybe that way we can find a workaround that doesn't make MS's shader compiler go nuts :/

#21 - 12/02/2012 09:20 PM - limburgerite

Changed the only line in shader_016.txt that looked like that to
crastemp = rastemp; //frac(rastemp * (255.0f/256.0f)) * (256.0f/255.0f);
but no dice, unfortunately... (success with /Od, crash without)

#22 - 12/04/2012 03:31 PM - limburgerite

Well, I finally have a working, up-to-date git/Dolphin build environment.

I've built two versions, one directly from r47aaca89eb0d and another from r47aaca89eb0d with rbb8b5936c040 reverted.

The latter does fix the crashing, so I've attached the problem shader from both versions. They differ by four lines, and adding the following line from the reverted version's shader to the unreverted one:

```
prev = frac(4.0f + prev * (255.0f/256.0f)) * (256.0f/255.0f);
```

stops the crashing (the other three lines have no effect).

Commenting out the following line:

<http://code.google.com/p/dolphin-emu/source/browse/Source/Core/VideoCommon/Src/PixelShaderGen.cpp?r=47aaca89eb0d36d307a9ac1f76293b9e5c9de257#760>

achieves roughly the same effect by forcing the "prev = frac(..." line to be written (though without the "4.0f + " part, which I guess is from the reverted code)

I doubt this is a proper fix, but hopefully it'll help you find a workaround...

#23 - 12/13/2012 08:05 AM - timbudtwo

I am having the same issue. How do I fix this or has there been a release with the fix already?

#24 - 12/13/2012 10:34 AM - scientificraver

Dolphin seems to be in a very unstable state at the moment so using an older revision like 3.0-750 until the relevant bugs are fixed is the only solution for the moment.

#25 - 12/13/2012 05:16 PM - limburgerite

Actually, this is likely the fault of a bug in Microsoft's shader compiler. You can switch to OpenGL to get through a realm then switch back to D3D after you've completed it.

#26 - 12/13/2012 05:24 PM - NeoBrainX

limburgerite: So let me get this right, commenting out this single line
<http://code.google.com/p/dolphin-emu/source/browse/Source/Core/VideoCommon/Src/PixelShaderGen.cpp#755> fixes the issue?

#27 - 12/13/2012 07:02 PM - limburgerite

Yep, just tried it with a clean, up-to-date master and just that one change makes the crashes go away. Looks like forcing the "emulation of unsigned 8 overflow" makes the compiler's optimizer happy. $\text{¯}(\text{°}_o)\text{¯}$

Plus, setting the D3D10_SHADER_SKIP_OPTIMIZATION flag (dunno the DX9 equivalent) also stops the crashing without needing the above fix, so at this point I'm 99% certain the problem is with the optimizer.

#28 - 12/13/2012 10:52 PM - NeoBrainX

Mhm I noticed something weird about the shader_16.txt you posted and changed some code around in the pixel shader gen.

Could you apply the patch from <http://pastie.org/5522428> on vanilla Dolphin and report back your results? I hope you don't get any shader compile errors (couldn't really test it locally). If you still get crashes, it would be cool if you uploaded the shader where it crashes at, like you did before.

#29 - 12/13/2012 10:53 PM - NeoBrainX

lol sorry, need to correct the patch already:
In patch line 62, it should be
"WRITE(p, "%s", tevOpTable[ac.op]);" (i.e. ac.op instead of cc.op)

#30 - 12/13/2012 11:17 PM - limburgerite

patch applied successfully to vanilla master (along with cc->ac correction) and...

:(

Debug info (ps_5_0):
(56,10): warning X3206: implicit truncation of vector type
internal error: compilation aborted unexpectedly

#31 - 12/13/2012 11:21 PM - NeoBrainX

Ok guess I screwed something up, but <http://pastie.org/5522529> should fix that. I also fixed the ac.op<->cc.op thing from above.

#32 - 12/13/2012 11:23 PM - NeoBrainX

Should fix the warning anyway... doubt it'll fix the actual issue now though, but try anyway and post the shader again :)

#33 - 12/13/2012 11:32 PM - limburgerite

oops...

Debug info (ps_3_0):

C:\Users\shmoo\Desktop\Dolphin\bin\memory(41,19): error X3004: undeclared identifier 'textempfloat4'

#34 - 12/13/2012 11:35 PM - NeoBrainX

... my fault, sorry :D

<http://pastie.org/5522571>

#35 - 12/13/2012 11:45 PM - limburgerite

This does fix the crash, but introduces some undesirable side effects (eg, some things that should be transparent aren't and things that shouldn't be are). I'll try to get some screenshots.

#36 - 12/13/2012 11:45 PM - NeoBrainX

Mhm, no need for screenshots, I guess I'll have to review my changes again since there's likely some mistake I made... :)

#37 - 12/13/2012 11:49 PM - NeoBrainX

... not today, however. Thanks again for testing though!

#38 - 12/14/2012 12:15 AM - limburgerite

No prob, glad to help...

Here's a screenshot anyway for the heck of it:

http://s9.postimage.org/ioc4h8f3/SOUE01_2.jpg

(note the lack of transparency in the save file textures)

#39 - 12/17/2012 06:49 PM - NeoBrainX

Finally found what was wrong about the previous patch:

<http://pastie.org/5543904>

Now let's hope the shader compiler is still fine with that one... (the glitches you reported should be gone for sure now)

#40 - 12/17/2012 10:41 PM - limburgerite

Visuals are back to normal, but sadly, still crashing at
D3DCompiler_43.dll!CProgram::SplitRegisters() + 0x21d bytes

Just out of curiosity, I did some tests forcing Dolphin to use other D3DCompiler_xx.dll versions, and this is what I came up with:

33, 34, 37 work

35, 36, 41, 42, 43, 46 crash (46 is the Windows 8 SDK version)

40 works with DX9 but crashes with DX11

(versions < 40 don't support ps_5_0, so they were only tested w/DX9)

Not sure it's of any use, but at least you know there are some versions that will actually compile the problem shaders.

#41 - 12/17/2012 10:52 PM - NeoBrainX

Could you try compiling in D3D11 with the most recent d3dcompiler.dll but using the D3DCOMPILER_OPTIMIZATION_LEVEL0, D3DCOMPILER_OPTIMIZATION_LEVEL1, D3DCOMPILER_OPTIMIZATION_LEVEL2 and D3DCOMPILER_OPTIMIZATION_LEVEL3 flags ? (each on its own, ofc)

#42 - 12/17/2012 10:54 PM - NeoBrainX

Note that we're currently using optimization level 3, so you'll want to drop that from the existing flags.

Also, could you add D3DCOMPILER_WARNINGS_ARE_ERRORS and/or D3DCOMPILER_ENABLE_STRICTNESS? I doubt they will be helping because they'll likely point at other parts of the code, but there's a small chance they do actually help.

Anyway, playing around with the opt. level is more important atm :->

#43 - 12/17/2012 10:59 PM - NeoBrainX

Uh, you'll want to edit the file Source/Plugins/Plugin_VideoDX11/Src/D3DShader.cpp at lines 49/51, 106/108 and 165/167 for that.

#44 - 12/17/2012 11:10 PM - limburgerite

Heh, optimization levels were one of the first things I messed with. All three levels crash; only skipping optimization works. :)

I'll play around with the other flags and see how it goes...

#45 - 12/17/2012 11:12 PM - limburgerite

Er, I mean all *four* levels crash (0-3).

#46 - 12/17/2012 11:33 PM - limburgerite

Nope, no change with any combo of D3DCOMPILER_WARNINGS_ARE_ERRORS and/or D3DCOMPILER_ENABLE_STRICTNESS (optimization flag was left out so should've defaulted to 1).

Are the 49/51 & 106/108 edits necessary? The crashing is only happening in CompilePixelShader().

#47 - 12/19/2012 02:30 PM - NeoBrainX

Nope, the other edits weren't really necessary.

Could you check that D3DXSHADER_SKIPOPTIMIZATION is the correct flag to use to workaround the issue in the d3d9 backend?

I think the only option we have right now is to disable optimizations altogether for pixel shaders :/

#48 - 12/19/2012 06:29 PM - NeoBrainX

For reference, this minimal shader makes the compiler crash already:

```
uniform float4 input;

void main(out float4 ocol0 : SV_Target0)
{
float4 temp = input.rgba / 2.f;
```

```
temp.a = saturate(temp.a);
temp.rgb = saturate(temp.rgb);

temp.rgb = lerp(temp.rgb, float3(0.5, 0.5f, 0.75f), float3(0.5f,0.5f,0.5f));
ocol0 = temp;
}
```

#49 - 12/19/2012 06:55 PM - NeoBrainX

<http://code.google.com/p/dolphin-emu/source/browse/Source/Core/VideoCommon/Src/PixelShaderGen.cpp#1329>

What if you revert all other changes you did before and change just that line to
WRITE(p, " prev = lerp(prev," I_FOG"[0],float4(fog.rrr,0.0f));\n");
?

(try this before trying anything else :D)

#50 - 12/19/2012 08:17 PM - limburgerite

Still crashes, but I think I see where you're going...

It looks like there's something weird going on with the .rgga. Specifically, it's having the middle two terms being the same (eg rbbg, grrb, rrrr all crash while rrar, bbag, argg don't).

In your minimal shader, splitting it up into:

```
float4 temp;
temp.rg = input.rg / 2.f;
temp.ba = input.ga / 2.f;
stops the crashing, and ironically seems to be optimized out anyway, even at level 0. Strange...
```

#51 - 12/19/2012 08:38 PM - limburgerite

Actually, I was a bit off about the patterns that crash, its more like:

```
abbc
(eg rrrr, gggg, rggg, gggg don't crash, but rgga and aggr do)
```

#52 - 12/19/2012 08:51 PM - NeoBrainX

Yeah, don't bother anymore I guess... we discussed this on irc quite long and found different "fixes" which work for some people and don't work for others. There does not seem to be a universal workaround :/

Can only hope MS will correct that bug in a future release.

#53 - 12/19/2012 08:54 PM - limburgerite

Yeah, I just tried my "fix" on the Dolphin-crashing shader and it didn't work :(

Don't forget to report this to MS:

#54 - 01/31/2013 10:56 AM - NeoBrainX

Gread, someone just spammed this issue and flooded everyone with >10 emails to post useless info on an issue that has been identified already.

Next time, upload crash traces and the like to a pastebin service and upload images somewhere else. Also, check if it's necessary at all to post any further info >_>

Efforts appreciated and stuff, though.

#55 - 02/01/2013 02:20 PM - scientificraver

I would not call it "not a Dolphin bug! It worked fine before, why not revert the commit that causes the error, even if it eventually seems to be a DirectX bug.

Of cause you can use OpenGL to avoid running into the issue right now, but that is slower on most setups.

#56 - 02/07/2013 06:06 AM - mnbayazit

I'm having this exact issue as well in ver 3.5-367. Switching to OpenGL does *not* fix the issue; in fact, it makes it worse. It crashes immediately after calibrating my Wiimote shortly after booting the game. I haven't found a way past this point in the game yet. The 3.0-win64 build didn't work either.

#57 - 02/19/2013 07:20 PM - scientificraver

I do not want to spam this thread any further but you can use 3.0-750 which is very stable in my opinion. Any revisions before 3.0-772 will not crash in DX9 or 11 when entering the trials.

#58 - 02/21/2013 07:21 PM - scientificraver

Maybe someone with an up to date Intel CPU might want to try this unofficial patched build:
<http://forums.dolphin-emu.org/Thread-dolphin-icc-intel-optimized-builds-sse3-4-avx-latest-3-5-419-x64-unofficial> (Dolphin 3.5-416 [Zelda-SS-patch] x64 ICC SSSE3,SSE4.1,SSE4.2,AVX) and report back to the forum if it fixes the crash or not
It reverts the 'fix' that is supposed to cause the SS silent realm crash.

#59 - 02/21/2013 07:25 PM - white.phoenix

I didn't revert the reverted revert, I just commented out line 755 of this file:
<http://code.google.com/p/dolphin-emu/source/browse/Source/Core/VideoCommon/Src/PixelShaderGen.cpp#755> which was reported to have fixed the issue. that said, I definitely am interested to see if this fix still works or not or if it has any unintended side effects.

#60 - 02/21/2013 07:25 PM - NeoBrainX

755 is the wrong line.

#61 - 02/21/2013 07:29 PM - white.phoenix

@ NeoBrainX: you asked above (in comment #29)if commenting out that line worked, and comment #30 said yes. did I misunderstand that or something? which line is it then?

#62 - 02/21/2013 07:31 PM - white.phoenix

Nevermind, I see now it's line 760, I'll fix my builds

#63 - 02/21/2013 07:34 PM - NeoBrainX

Still wrong; no idea what revision you're looking at, but I guess it's this line:

<http://code.google.com/p/dolphin-emu/source/browse/Source/Core/VideoCommon/Src/PixelShaderGen.cpp?r=29d43ef897275bd200cee0ec4492b1c2052902f3#707>

(either that, or that and line 708)

Anyway, commenting out any of this results in bad hardware emulation and thus obviously is no fix. If anything, you might want to try the tev_fixes branch. Maybe that one made the code complex enough to change the failure of the shader optimizer :p

#64 - 02/21/2013 07:55 PM - NeoBrainX

Oh hey, actually commenting out only line 707 would be a good fix. If it still fixes the issue, I'd be tempted to force line 708 to always be enabled :p

#65 - 02/21/2013 08:34 PM - white.phoenix

I got the right one this time, [#707](#)

```
//if(RegisterStates[0].AlphaNeedOverflowControl || RegisterStates[0].ColorNeedOverflowControl)
```

Build two builds, one ICC compiled and one standard vanilla MSVC w/ `_M_SSE=0x40` added to preprocessor for SSE3/SSE4.

Both are posted in my thread on the forums, but also adding them below in case anyone wants to try them.

Dolphin 3.5-420 [Zelda-SS-Fix] x64 ICC SSSE3,SSE4.1,SSE4.2,AVX + OpenMP <http://adf.ly/JZplc>

Dolphin 3.5-420 [Zelda-SS-Fix] x64 MSVC SSE3,SSE4 Intel+AMD <http://adf.ly/JZptE>

#66 - 02/21/2013 08:50 PM - white.phoenix

crap I just realized `_M_SSE=0x40` does SSSE3 which AMD doesn't have. Here's a straight vanilla build which will work on any CPU.

Dolphin 3.5-420 [Zelda-SS-Fix] x64 MSVC Intel+AMD <http://adf.ly/JZuKU>

Sorry for the email spam for those of you subscribed to this.

#67 - 02/21/2013 09:21 PM - scientificraver

My savegame is currently far away from the crash so I cannot test. However, reverting Revision: bb8b5936c040 which fixes a bug in WindWaker but initially caused the SS crash should fix the bug in theory.

#68 - 02/28/2013 03:33 AM - MattBoySlim

white.ph~-- I just encountered this crash tonight (after finally getting an overheating issue under control and being able to run SS in a playable state). I tried your patched ICC version and it definitely seems to be working...got me past the crash point, anyway. Thanks!

#69 - 03/02/2013 10:43 PM - scientificraver

I can confirm that the [Zelda-SS-Fix] builds do not crash when entering the trials in DX9.

#70 - 03/20/2013 04:32 AM - mnbayazit

I too can confirm that "Dolphin 3.5-420 [Zelda-SS-Fix] x64 MSVC Intel+AMD" fixes the issue. I backed up my "Wii" folder just before a silent realm so I should be able to test if you guys submit a fix to the official branch, but it looks like white.ph already has it nailed down. Can we expect this this to

make in into an official build anytime soon?

#71 - 04/10/2013 01:53 AM - Billiard26

- Issue type set to Bug

#72 - 05/17/2013 03:03 AM - skidau

[issue 5883](#) has been merged into this issue.

#73 - 05/17/2013 04:11 AM - white.phoenix

@mnbaya...@gmail.com: it's not really a dolphin bug, but a DirectX bug. Last I heard, dolphin was waiting on a fix from microsoft. My patch just reverts part of a commit that caused the bug to surface.

#74 - 06/02/2013 12:34 AM - fanfoisinc

This crash has happened to me before but I just tried 3.5-1386 and it just works now !
I double-checked and here is my now working situation :
Din's trial, with Direct3D11.
I7 3600k, GTX 560 Ti, 1080p, DSP LLE.
Can anyone confirm ?

#75 - 08/03/2013 11:22 AM - delroth

- Priority set to High

- Operating system Windows added

Does this still happen with recent revisions?

Need to get someone to ping MS again about this.

#76 - 08/11/2013 12:40 AM - Autoran1

[issue 6281](#) has been merged into this issue.

#77 - 08/20/2013 09:32 PM - Autoran1

Decided to replay this game again, and found this issue, yes the recent builds still have it, tried to revert part of rbb8b5936c040, as it were said here <http://code.google.com/p/dolphin-emu/issues/detail?id=5682#c86> and everything worked

#78 - 09/20/2013 07:11 PM - velocity7

Issue still exists as of 3.5-2436, Direct3D 9. Easiest place to try this is actually in Lanaryu Gorge; talk to the Thunder Dragon to start a Silent Realm trial.

#79 - 09/30/2013 12:05 AM - squallfer

issue still on 4.0 :S but in 3.0-255 work's great

#81 - 01/21/2014 10:15 AM - NeoBrainX

Not a problem in tev_fixes_new.

#82 - 03/14/2014 10:24 PM - delroth

- Status changed from Accepted to Fixed

Fixed in commit a9a8c730748b (tev_fixes_new).