

Subject: : Odyssey

Topic: : Odyssey 1.23 progress: r5 beta07

Re: Odyssey 1.23 progress

Author: : kas1e

Date: : 2020/2/28 18:56:11

URL:

@all

Source code uploaded, check this out:

https://www.amigans.net/modules/xforum...hp?topic_id=8179&forum=40

Also, there was some progress I do lately: I rebuild the whole beast over GCC 8.2.0 and with fixing some annoying bits I was lucky enough to make it compiles and links (that was a side effect of making preparation for GitHub upload).

Switching to 8.x GCC may give us more speed, as well as give us some warning about things that need to be fixed. It is also some kind of pre-test to porting later 1.25 version from Deadwood's repo.

Binary works, though it did bring me some scary new bug that needs to be sorted out before another beta-release.

New Issue is that for some reason paths in some parts are all trashed now. For example, PROGDIR:Conf/ for saving/loading webicons.db, have "Conf/" trashed with all weird characters of a strange size. Building the same code over GCC 4.4.3 (with some small changes not related to that issue) does not have that bug.

Visually when you start new binary, you have at first no webicons, as they can't be loaded from PROGDIR:Conf/WebPagelcons.db, and instead created a new directory of messed characters in which place new WebPagelcons.db

All I can say about the bug is that it comes from "threads". For example, there is an IconDataBase thread, which creates WebIcons.db. Or the same happens with the "LocalStorage" thread, which saves things too and which "path" is messed now.

As with web icons it easier to trace, I just checked that it loaded from there:

<https://github.com/kas1e/Odyssey/blob/...con/IconDatabase.cpp#L926>

So, it's defaultDatabaseFilename somehow trashes.

Checking then where that defaultDataBaseFilename setup, there are functions which init all that:

<https://github.com/kas1e/Odyssey/blob/...i/WebIconDatabase.cpp#L96>

~~So any actual help (taking into account that all can browse source code already) appreciated!~~

I was sure at first it's something about "athread=native", but then switching to "athread=single" (so be like old gcc 4.4.3) do not solve the issue, sadly.