Subject: : AmigaOS4
Topic: : SDL2
Re: SDL2
Author: : kas1e
Date: : 2019/3/30 20:22:59
URL:

@Capehill
Quote:

1) Does Lugaru call SDL_GL_SetAttribute?

Yes, that is how initialisation done before we do call to that failing code:

```
if (!SDL_WasInit(SDL_INIT_VIDEO)) {
    if (SDL_Init(SDL_INIT_VIDEO) == -1) {
        fprintf(stderr, "SDL_Init() failed: %sn", SDL_GetError());
        return false;
    }
}
if (!LoadSettings()) {
    fprintf(stderr, "Failed to load config, creating defaultn");
    SaveSettings();
}

if (SDL_GL_LoadLibrary(NULL) == -1) {
    fprintf(stderr, "SDL_GL_LoadLibrary() failed: %sn", SDL_GetError());
    SDL_Quit();
    return false;
}

for (int displayIdx = 0; displayIdx < SDL_GetNumVideoDisplays(); ++displayIdx) {
    for (int i = 0; i < SDL_GetNumDisplayModes(displayIdx); ++i) {
        SDL_DisplayMode mode;
        if (SDL_GetDisplayMode(displayIdx, i, &mode) == -1) {
            continue;
        }
        if ((mode.w < 640) || (mode.h < 480)) {
```

```
            continue; // sane lower limit.
        }
        pair<int, int> resolution(mode.w, mode.h);
        resolutions.insert(resolution);
    }
  }

  if (resolutions.empty()) {
    const std::string error = "No suitable video resolutions found.";
    cerr << error << endl;
    SDL_ShowSimpleMessageBox(SDL_MESSAGEBOX_ERROR, "Lugaru init failed!", error.c_str(), NULL);
    SDL_Quit();
    return false;
  }

  if (commandLineOptions[SHOWRESOLUTIONS]) {
    printf("Available resolutions:n");
    for (auto resolution = resolutions.begin(); resolution != resolutions.end(); resolution++) {
      printf("  %d x %dn", (int)resolution->first, (int)resolution->second);
    }
  }

  SDL_GL_SetAttribute(SDL_GL_DOUBLEBUFFER, 1);
  SDL_GL_SetAttribute(SDL_GL_STENCIL_SIZE, 1);

  Uint32 sdlflags = SDL_WINDOW_OPENGL | SDL_WINDOW_SHOWN;
  if (commandLineOptions[FULLSCREEN]) {
    fullscreen = commandLineOptions[FULLSCREEN].last()->type();
  }
  if (fullscreen) {
    sdlflags |= SDL_WINDOW_FULLSCREEN;
  }
  if (!commandLineOptions[NOMOUSEGRAB].last()->type()) {
    sdlflags |= SDL_WINDOW_INPUT_GRABBED;
  }

  sdlwindow = SDL_CreateWindow("Lugaru", SDL_WINDOWPOS_CENTERED_DISPLAY(0),
SDL_WINDOWPOS_CENTERED_DISPLAY(0),
                               kContextWidth, kContextHeight, sdlflags);

  if (!sdlwindow) {
    fprintf(stderr, "SDL_CreateWindow() failed: %sn", SDL_GetError());
    fprintf(stderr, "forcing 640x480...n");
    kContextWidth = 640;
    kContextHeight = 480;
    sdlwindow = SDL_CreateWindow("Lugaru", SDL_WINDOWPOS_CENTERED_DISPLAY(0),
SDL_WINDOWPOS_CENTERED_DISPLAY(0),
                                 kContextWidth, kContextHeight, sdlflags);
    if (!sdlwindow) {
      fprintf(stderr, "SDL_CreateWindow() failed: %sn", SDL_GetError());
      fprintf(stderr, "forcing 640x480 windowed mode...n");
```

```
        sdlflags &= ~SDL_WINDOW_FULLSCREEN;
        sdlwindow = SDL_CreateWindow("Lugaru", SDL_WINDOWPOS_CENTERED_DISPLAY(0),
SDL_WINDOWPOS_CENTERED_DISPLAY(0),
                                     kContextWidth, kContextHeight, sdlflags);

        if (!sdlwindow) {
            fprintf(stderr, "SDL_CreateWindow() failed: %sn", SDL_GetError());
            return false;
        }
    }
  }

  SDL_GLContext glctx = SDL_GL_CreateContext(sdlwindow);
  if (!glctx) {
    fprintf(stderr, "SDL_GL_CreateContext() failed: %sn", SDL_GetError());
    SDL_Quit();
    return false;
  }

  SDL_GL_MakeCurrent(sdlwindow, glctx);

  int dblbuf = 0;
  if ((SDL_GL_GetAttribute(SDL_GL_DOUBLEBUFFER, &dblbuf) == -1) || (!dblbuf)) {
    fprintf(stderr, "Failed to get a double-buffered context.n");
    SDL_Quit();
    return false;
  }
```

As you can see, before call we have "int dblbuf = 0;".

What is strange, is that it happens only with minigl build. With gl4es build it didn't happens (but, gl4es in my hacked SDL its just replaced minigl parts mostly, through some functions a bit different).

Also the fact that the same code works on all other platforms (win32,linux, etc) make me think it can be something in our SDL.

Quote:

> 2) what is the error, SDL_GL_GetAttribute returning negative or dblfbuf being zero?

Being zero.

I may try to create simple test case from the code above, and do check it on win32 for example to see how it reacts.