
Subject: : AmigaOS4

Topic: : MineCraft (MineTest) work in progress help need it

Re: MineCraft (MineTest) work in progress help need it

Author: : kas1e

Date: : 2021/1/13 9:03:19

URL:

@All

In end i think i better go the "amiga paths" way, i.e. removing -linux from linking line and fix amiga paths.

There from begining bunch of little problems:

1. rename() function behave differently

-- it didn't delete an exiting file with the same name to which we tried to rename

-- on posix 0 is return if succes, on amigados 0 is return if NOT succes.

That how i deal with:

```
#ifdef __amigaos4__
// AmigaDOS's Rename didn't overwrite an exiting destination file, so we mimic unix way: where rename() als
o delete a existing file with the same name.
// On linux if rename() succes 0 is return, on AmigaDOS 0 return if rename is failed instead. So we spaw a log
ic a bit:
IDOS->Delete(path.c_str());
rename_success = IDOS->Rename(tmp_file.c_str(), path.c_str()) != 0;

if (!rename_success) {
warningstream << "Failed to write to file: " << path.c_str() << std::endl;
// Remove the temporary file because moving it over the target file
// failed.
remove(tmp_file.c_str());
return false;
}
#else
rename_success = rename(tmp_file.c_str(), path.c_str()) == 0;
if (!rename_success) {
warningstream << "Failed to write to file: " << path.c_str() << std::endl;
// Remove the temporary file because moving it over the target file
// failed.
remove(tmp_file.c_str());
return false;
}
```

```
}  
#endif
```

2. There were some "double slashes" in the pathes, causing by the lua scripts in the parts which load menu's textures. I think at first to fix it in lua scripts, but fixing data files kind of suck, so instead i add in the menu loading textures functions that:

```
#ifdef __amigaos4__  
// function to replace X amount of bytes in the buffer on Y amount of byte  
std::string ReplaceAll(std::string str, const std::string& from, const std::string& to) {  
    size_t start_pos = 0;  
    while((start_pos = str.find(from, start_pos)) != std::string::npos) {  
        str.replace(start_pos, from.length(), to);  
        start_pos += to.length(); // Handles case where 'to' is a substring of 'from'  
    }  
    return str;  
}  
#endif
```

And then, in the int ModApiMainMenu::_set_background(lua_State *L)

.....bablla

```
if (!lua_isnone(L, 4)) {  
    minsize = lua_tonumber(L, 4);  
}
```

```
#ifdef __amigaos4__
```

```
// fix "/" in the paths.  
// when compiling withot -linux (for use amigados native paths), then "/" things will handled wrong.  
// so find out if we had "/" in the path and replace it with one "/"
```

```
texturename = ReplaceAll(texturename, std::string("/"), std::string("/"));
```

```
printf(" after fix texturename = %sn",texturename.c_str());
```

```
#endif
```

```
if (backgroundlevel == "background") {  
    retval |= engine->setTexture(TEX_LAYER_BACKGROUND, texturename,  
        tile_image, minsize);  
}
```

..... blalba...

3. RecursiveDelete() function (to delete created games) were written for win32 and for posix (with fork + rm -rf), so i go win32 way:

<https://github.com/minetest/minetest/blob/master/src/filesys.cpp#L126>

And replace it like that

```
#ifdef __amigaos4__

// Delete() on AmigaDOS return 0 if fail, and DeleteFile() on win32 return 0 if success, so change it as well.

bool RecursiveDelete(const std::string &path)
{
    infostream << "Recursively deleting " << path << "" << std::endl;
    if (!IsDir(path)) {
        infostream << "RecursiveDelete: Deleting file " << path << std::endl;
        if (IDOS->Delete(path.c_str())) {
            errorstream << "RecursiveDelete: Failed to delete file "
                << path << std::endl;
            return false;
        }
        return true;
    }
    infostream << "RecursiveDelete: Deleting content of directory "
        << path << std::endl;
    std::vector<DirListNode> content = GetDirListing(path);
    for (const DirListNode &n: content) {
        std::string fullpath = path + DIR_DELIM + n.name;
        if (!RecursiveDelete(fullpath)) {
            errorstream << "RecursiveDelete: Failed to recurse to "
                << fullpath << std::endl;
            return false;
        }
    }
    infostream << "RecursiveDelete: Deleting directory " << path << std::endl;
    if (IDOS->Delete(path.c_str())) {
        errorstream << "Failed to recursively delete directory "
            << path << std::endl;
        return false;
    }

    return true;
}

#endif
```

Not sure how correct is that, but works.

So, now we can run without -linux, and all seems loads correctly, no requests, games can be created/deleted/etc.