

---

Subject: : AmigaOS4

Topic: : Porting to AmigaOS4 thread

Re: Porting to AmigaOS4 thread

Author: : Raziel

Date: : 2020/11/22 11:49:45

URL:

@jabirulo

Thank you very much

I now get only one more warning

```
C++ backends/dialogs/amigaos/amigaos-dialogs.o
backends/dialogs/amigaos/amigaos-dialogs.cpp: In member function
'virtual Common::DialogManager::DialogResult AmigaOSDialogManager::showFileBrowser(const Common::U3
2String&, Common::FSNode&, bool)':
backends/dialogs/amigaos/amigaos-dialogs.cpp:111:56: warning: cast from type 'const value_type*' {aka
'const char*'} to type 'char*' casts away qualifiers [-Wcast-qual]
    char *newTitle = utf8ToLocal(((char *)utf8Title.c_str()));
                                   ^
```

- i get a working exe with an asl requester opening

- closing the requester works too, but

after closing the requester once i can not open another instances

Can you spot the mistake?

```
struct Library *AslBase = IExec->OpenLibrary(AslName, 50);
struct AslIFace *IAsl;
```

```
Common::DialogManager::DialogResult AmigaOSDialogManager::showFileBrowser(const Common::U32String
&title, Common::FSNode &choice, bool isDirBrowser) {
```

```
    char pathBuffer[MAXPATHLEN];
    Common::String utf8Title = title.encode();
    DialogResult result = kDialogCancel;
```

```
    if (AslBase) {
```

```

IAsl = (struct AslIFace*)IExec->GetInterface(AslBase, "main", 1, NULL);

struct FileRequester *fr = NULL;

if (ConfMan.HasKey("browser_lastpath")) {
    strncpy(pathBuffer, ConfMan.get("browser_lastpath").c_str(), sizeof(pathBuffer)-1);
}

fr = (struct FileRequester *)IAsl->AllocAslRequestTags(ASL_FileRequest, TAG_DONE);

if (!fr)
    return result;

char *newTitle = utf8ToLocal(((char *)utf8Title.c_str()));

if (IAsl->AslRequestTags(fr, ASLFR_TitleText, newTitle, ASLFR_RejectIcons, TRUE, ASLFR_InitialDrawer
, pathBuffer, ASLFR_DrawersOnly, (isDirBrowser ? TRUE : FALSE), TAG_DONE)) {

    if (strlen(fr->fr_Drawer) < sizeof(pathBuffer)) {
        strncpy(pathBuffer, fr->fr_Drawer, sizeof(pathBuffer));
        if (!isDirBrowser) {
            IDOS->AddPart(pathBuffer, fr->fr_File, sizeof(pathBuffer));
        }
        choice = Common::FSNode(pathBuffer);
        ConfMan.set("browser_lastpath", pathBuffer);
        result = kDialogOk;
    }
    free(newTitle);
    IAsl->FreeAslRequest((APTR)fr);
}
IExec->DropInterface((struct Interface*)IAsl);
IAsl = NULL;
IExec->CloseLibrary(AslBase);
AslBase = NULL;
}

return result;
}

```