

MUI-AmigaE

| |
|----------------------|
| COLLABORATORS |
|----------------------|

| | | |
|------------------|------------------------------|---------------|
| | <i>TITLE :</i> MUI-AmigaE | |
| <i>ACTION</i> | <i>NAME</i> | <i>DATE</i> |
| WRITTEN BY | | July 25, 2024 |
| <i>SIGNATURE</i> | | |

| |
|-------------------------|
| REVISION HISTORY |
|-------------------------|

| NUMBER | DATE | DESCRIPTION | NAME |
|--------|------|-------------|------|
| | | | |

Contents

| | | |
|----------|---|----------|
| 1 | MUI-AmigaE | 1 |
| 1.1 | MUI-AmigaE.guide | 1 |
| 1.2 | First words | 1 |
| 1.3 | Changes | 2 |
| 1.4 | The Files | 2 |
| 1.5 | Files: Modules/muimaster.m | 2 |
| 1.6 | Files: Modules/libraries/mui.e | 2 |
| 1.7 | Files: Modules/libraries/muip.e | 3 |
| 1.8 | Files: Modules/libraries/diffs.readme | 3 |
| 1.9 | Files: Examples/MUI-Demo.e | 3 |
| 1.10 | Install | 4 |
| 1.11 | Problems with MUI and E | 4 |
| 1.12 | Problems: TRUE=1 or TRUE=-1 | 4 |
| 1.13 | Problems: Problem with SetAttrsA() | 4 |
| 1.14 | Problems: TAG_IGNORE,0 | 5 |
| 1.15 | Author | 6 |
| 1.16 | Future | 6 |
| 1.17 | Disclaimer | 6 |

Chapter 1

MUI-AmigaE

1.1 MUI-AmigaE.guide

MUI DEVELOPER FILES FOR AMIGA E

~~~~~

|             |                               |
|-------------|-------------------------------|
| First words | IMPORTANT! READ THIS!         |
| Changes     | What's new ?                  |
| Files       | Description of the files.     |
| Install     | Install the files.            |
| Problems    | Problems with MUI and AmigaE. |
| Future      | New MUI -> new files ?        |
| Author      | and copyright.                |
| Disclaimer  | No warranty!                  |

### 1.2 First words

FIRST WORDS

~~~~~

I moved (from Passau to Hamburg)! Because of that, I lost my old e-mail address (schulzj@fmi.uni-passau.de) and my snail-mail address has changed too.

I try to get a new e-mail address as soon as possible, but if you read this, I didn't had this new e-mail address as this MUI version was ready for distributing. So, this Amiga-E files are maybe not 100% uptodate. Have a look at the file Modules/libraries/diffs.readme to see, which struct's and #defines of the original mui.h are not in the Amiga-E file. If necessary, I will distribute a "muiXXEdev.lha" with the completed Amiga-E files ASAP.

There are some other important changes , have a look at them!

1.3 Changes

CHANGES
~~~~~

Changes from "mui23Edev.lha":

- E versions <3.1a are no longer supported to keep the total size of this Amiga-E files reasonable. If you still use E v2.1b than update now or use "mui23Edev.lha".
- The modules in "Modules/libraries" are partly generated automaticly by Stefan Stuntz. Because he has no e-compiler, and to keep the number of files small, only the source-code of this modules is provided. You have to compile them into an e-module by yourself. (see Install )

## 1.4 The Files

THE FILES  
~~~~~

```
Modules/          muimaster.m

Modules/libraries/  mui.e
                   muip.e
                   diffs.readme

Examples/         MUI-Demo.e
```

1.5 Files: Modules/muimaster.m

"Modules/muimaster.m"
~~~~~

This file defines the functions of the muimaster.library. Copy it to EMODULES: and to use it write in your source:

```
MODULE 'muimaster'
```

ATTENTION: With "mui22Edev2.lha" I changed the names of the functions from MuI\_... to Mui\_... (Lionel Vintenat (Mac2E) told me that Mui\_... looks nicer and I think he's right.) Maybe you have to change your old sources to use this file.

## 1.6 Files: Modules/libraries/mui.e

"Modules/libraries/mui.e"  
~~~~~

This is the source-file of the e-module "mui.m". This module contains a complete translation of the original C mui.h file with only one exception:

The muip_... OBJECTS are moved to the file Modules/libraries/muip.e .

The #defines (and CONSTs) are written exactly the same way as in the mui.h file (with one exception: The macro String() is renamed to StringMUI() to avoid conflicts with the E function String()). The identifiers of the OBJECTS are all written totally lowercase!!

Compile this file with the e-compiler and copy the "mui.m" you get as result to "EMODULES:libraries/mui.m". To use this module write in your source:

```
MODULE 'libraries/mui'
```

Don't forget to add "OPT PREPROCESS" to your source, if you want to use the macros!

See also: Modules/libraries/muip.e

1.7 Files: Modules/libraries/muip.e

```
"Modules/libraries/muip.e"
~~~~~
```

This is the source for the e-module "muip.m". It contains all the MUIP_...-structs of the original C mui.h file as OBJECT's. I think, this OBJECT's are not very usefull in E, that's why I didn't include them in the first releases. But to have a complete AmigaE-MUI-Interface...

Compile this file with the e-compiler and copy the "muip.m" you get as result to "EMODULES:libraries/muip.m". To use this module write in your source:

```
MODULE 'libraries/muip.m'
```

1.8 Files: Modules/libraries/diffs.readme

```
"Modules/libraries/diffs.readme"
~~~~~
```

This file contains a list of all struct's and #defines of the mui.h file, which are not translated into E. (See First words for the reason!)

If you need one of them, look at the mui.h file, or wait for an update of this AmigaE-files (available as "dev/e/muiXXEdev.lha" on aminet).

I hope this list is not too long! :-|

1.9 Files: Examples/MUI-Demo.e

```
"Examples/MUI-Demo.e"
~~~~~
```

This is a translation of the C demo-source "MUI-Demo.c" that came with MUI v2.1. Use it to see, how to write MUI-programms in E and where are the differences between C and E.

See also: Problems

1.10 Install

```
INSTALL
~~~~~
```

To install the AmigaE-MUI-files do the following:

1. Open a shell
2. CD into the 'Developer/Amiga-E/Modules' directory
3. execute 'COPY muimaster.m EMODULES:'
4. CD into the 'libraries' subdirectory
5. call the e-compiler with 'EC mui.e'
6. execute 'COPY mui.m EMODULES:libraries/mui.m'
7. call the e-compiler with 'EC muip.e'
8. execute 'COPY muip.m EMODULES:libraries/muip.m'

1.11 Problems with MUI and E

```
PROBLEMS WITH MUI AND E
~~~~~
```

```
MUI_TRUE      Why a CONST MUI_TRUE?
SetAttrsA()   Problems with set()
TAG_IGNORE    Why "TAG_IGNORE,0," in some macros?
```

1.12 Problems: TRUE=1 or TRUE=-1

```
TRUE=1 or TRUE=-1
~~~~~
```

In C TRUE has the value 1 but in E TRUE has the value -1. That can cause problems, therefore a CONST "MUI_TRUE" with value 1 is defined in Modules/libraries/mui.e . Use this one instead of TRUE whenever you want to give TRUE to MUI. See "Examples/MUI_Demo.e".

1.13 Problems: Problem with SetAttrsA()

Problem with SetAttrsA()
 ~~~~~

Whenever you try to set an attribut to the value it allready has, MUI overwrites this attribut in the taglist with TAG\_IGNORE to make shure that notify-class don't react on it. This is nessesary to prevent endless-notification-loops. As long as you are a C programmer and use SetAttr() that is no problem, because MUI then changes only a \*copy\* of the original datas of the SetAttrs()-call. But if you use SetAttrsA(), as we must do because SetAttrs() is a function of amiga.lib, it is a problem! You only give a PTR to the original-datas with SetAttrsA() and now MUI changes this \*original\* datas. If you then uses this datas again at the next execution of the same SetAttrsA()-call, the datas are still changed and nothing (TAG\_IGNORE) will happen :-)

One possible way to avoid this problem is, to make the attribut a non-constant data. Instead of

```
SetAttrsA(obj, [ MUIA_..., value, ..., TAG_DONE])
```

write

```
SetAttrsA(obj, [Eval(`(MUIA_...)), value, ..., TAG_DONE])
~~~~~      ~~
```

Now everytime this SetAttrsA()-call is executed, the attribute is again "evaluated" and stored in the list.

BUT: x:=[Eval(`(MUIA\_...)),...]; SetAttrsA(obj,x); SetAttrsA(obj,x) has \*NOT\* the wished result, because the non-constand datas of the LIST are only evaluated when x is assigned to to the LIST but not everytime this PTR to the LIST in x is used later!

To avoid the SetAttrsA()-problem the macros set() (and nnset()) in the file Modules/libraries/mui.e are defined like this:

```
#define set(obj,attr,value) SetAttrsA(obj,[Eval(`(attr)),value,TAG_DONE])
```

The global variable "setAttrsA" that was needed with the set() and nnset() of former releases is now obsolete.

## 1.14 Problems: TAG\_IGNORE,0

TAG\_IGNORE,0  
 ~~~~~

Some of the macros in Modules/libraries/mui.e are ending with "[TAG\_IGNORE,0,". That seems to be superfluous but it is needed, because in AmigaE v2.1b if you want to split a statement over several lines, you can't end a line with "[" but with a comma. And with AmigaE v3.1 it's needed to allow the same use of the macros.

## 1.15 Author

AUTHOR  
~~~~~

IMPORTANT NOTE: I moved and my address changed! (see First words )

Snail mail (new): Jan Hendrik Schulz  
Lewetzauweg 16  
22455 Hamburg  
Germany

e-mail: no e-mail address available now!

## 1.16 Future

FUTURE  
~~~~~

I try to get a new e-mail address ASAP, and I hope the next MUI version will come with an uptodate AmigaE-Interface...

If you have some ideas to make things better, please write me .

## 1.17 Disclaimer

DISCLAIMER  
~~~~~

This files are provided "AS IS" without warrenty of any kind, expressed or implied! I'm NOT liable to you for damages or problems, including any general, special, incidental or consequential damages or problems arising out of the use or inhability to use of the files. Including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of a program build with this files.

(I hope you know what I want to say! My english is not good enough to write such legal stuff!)

---