

**mui22Edev2.readme**

**COLLABORATORS**

	<i>TITLE :</i> mui22Edev2.readme		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		October 17, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>mui22Edev2.readme</b>	<b>1</b>
1.1	mui22Edev2.README . . . . .	1
1.2	Introduction . . . . .	2
1.3	Changes . . . . .	2
1.4	Filename extensions . . . . .	3
1.5	The Files . . . . .	3
1.6	Files: ReadMe.guide . . . . .	4
1.7	Files: Modules/muimaster.m . . . . .	4
1.8	Files: Modules/libraries/mui.m . . . . .	5
1.9	Files: Modules/libraries/Ev3_mui.m . . . . .	5
1.10	Files: Modules/libraries/muip.m . . . . .	6
1.11	Files: Modules/libraries/Ev3_muip.m . . . . .	6
1.12	Files: MacroFiles/mui.ma . . . . .	7
1.13	Files: MacroFiles/Ev3_mui.ma . . . . .	7
1.14	Files: MacroFiles/mui.pma . . . . .	8
1.15	Files: MacroFiles/Ev3_mui.pma . . . . .	8
1.16	Files: Sources/doMethod.e . . . . .	8
1.17	Files: Sources/installhook.e . . . . .	9
1.18	Files: Sources/MUI-Demo.em . . . . .	9
1.19	Files: Sources/MUI-Demo.e . . . . .	9
1.20	Files: Sources/PopUp.em . . . . .	10
1.21	Files: Sources/PopUp.e . . . . .	10
1.22	Problems with MUI and E . . . . .	10
1.23	Problems: TRUE=1 or TRUE=-1 . . . . .	11
1.24	Problems: Problem with SetAttrsA() . . . . .	11
1.25	Problems: TAG_IGNORE,0 . . . . .	12
1.26	Copyrights and authors . . . . .	12
1.27	My address . . . . .	13
1.28	Future . . . . .	13
1.29	Bugs . . . . .	13
1.30	Disclaimer . . . . .	14

---

# Chapter 1

## **mui22Edev2.readme**

### **1.1 mui22Edev2.README**

```
MUI2.2 DEVELOPER FILES FOR AMIGA E
~~~~~
```

Introduction

What is mui22Edev2.lha ?

Changes

What's new ?

Files

Description of the files.

Problems

Problems with MUI and AmigaE.

Future

New MUI -> new files ?

Bugs

Copyrights

and authors.

Disclaimer

No warranty!

I'm wondering if there is anybody using this files. Until now I got only one reaction (from Lionel Vintenat (Mac2E)). So, if you use this files, please think about sending

me

an e-mail (or maybe a postcard). You don't have to write a long letter, only something like: "I use your mui22Edev2-files". Thankyou!

Jan Hendrik Schulz

---

## 1.2 Introduction

### INTRODUCTION

~~~~~

The original files for Amiga-E programmer that came with the developer archive of

MUI

until now are not very usefull.

The muimaster.m file, that should define all the functions of the muimaster.library, is not up-to-date, and the mui.e file is only a very bad replacement for the mui.h file that C programmer could use!

Thats why I created my own MUI-AmigaE-interface. This archive (as a replacement for the original Amiga-E drawer of the MUI developer archive) contains nearly all what's needed to write MUI-programs in E and some additional informations in this file! The only "extra" you need (to use the macrofile) is the program

Mac2E

.

## 1.3 Changes

### CHANGES

~~~~~

Changes from "mui22Edev.lha" to "mui22Edev2.lha"

- Readme-file is now an amigaguide dokument.
- Now ALL struct's of the original mui.h file as OBJECT's in modules available. (see
  - muip.m
  - )
- Special module- and macro-files for AmigaE v3
  - (
  - Ev3\_mui.m
  - ,
  - Ev3\_muip.m
  - ,
  - Ev3\_mui.ma
  - and
  - Ev3\_mui.pma
  - )
- The names of the functions in muimaster.m are changed from MuI\_... to Mui\_...!
- Some files removed:
  - "installhook2.e" and "installhook2.m", because it's possible to use the normal
    - installhook
    - instead.
  - "domethod.m", because it's available as "tools/Boopsi.m" with the AmigaE v3 distribution.

- MUIMASTER\_VMIN in  
     mui.m  
     changed from 8 to 9 (In the original  
     mui.h file of mui22dev.lha it's 8, but that's probably wrong!?)
- New set() and nnset() macro definitions -> No setAttrSA variable  
     needed. (Different from the Mac2E v4 definition, see:  
     new set()  
     )

Changes from "mui21Edev.lha" to "mui22Edev.lha"

- New modules for MUI 2.2 (Only "modules/libraries/mui.m" is new)
- New filename-extensions to give a better indication of the filetype
- Some new files
- Sourcecodes etc. changed to use tabs instead of spaces where it's  
     possible, to decrease filesize. (Use an editor or textviewer with  
     tabstops at every 8th column to get correct textformatting!)

## 1.4 Filename extensions

### FILENAME-EXTENSIONS

~~~~~

- .e - An E-sourcecode that could be compiled directly.
- .m - An E-module file.
- .em - An E-Sourcecode with some macros.  
     Mac2E  
     is needed to  
     create the E-Sourcecode (".e") for the E-compiler EC.
- .ma - This is a macrofile.
- .pma - And this is a pre-analysed macrofile

## 1.5 The Files

### THE FILES

~~~~~

```

ReadMe.guide
Modules/
muimaster.m
Modules/libraries/
mui.m

Ev3_mui.m

muip.m

Ev3_muip.m
MacroFiles/
mui.ma

```

```

Ev3_mui.ma

mui.pma

Ev3_mui.pma
Sources/
domethod.e

installhook.e

MUI-Demo.em

MUI-Demo.e

PopUp.em

PopUp.e
Information about the
Filename-Extensions

```

## 1.6 Files: ReadMe.guide

```

"ReadMe.guide"
~~~~~

This file. Please read the entire file before mailing
me
something like:
"This files are creating only errors, what make I wrong?" There are some
special
problems
with MUI and Amiga-E which can cause big trouble :- (

```

## 1.7 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 this release 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.) Sorry, you have to change your old sources to use this new file (which is required by the macrofiles!), but it's not that complicate. Simply replace all "MuI\_" by "Mui\_" with the replace-function of your texteditor.

---

## 1.8 Files: Modules/libraries/mui.m

```

"Modules/libraries/mui.m"
~~~~~

```

This file contains some of the #define's and struct's of the mui.h file that came for C programmer with the developer archive of MUI 2.2. Use the C mui.h file with its comments to get more informations about the #define's and struct's.

In this E-module-file all CONSTs and the structs (OBJECTs) which are of interest for E programmer, are defined. To see which structs are in this module, use the ShowModule-tool that came with the E compiler. Therefore all OBJECTs are at the beginning of the module.

The CONSTs are written exactly the same way as in the mui.h file (with lowercase letters in them - that is no problem as long as the two first are uppercase, even if the E-language-description says that CONSTs have to be totally uppercase!) So it's no problem to use this files even if you had used the macrofile that came with Mac2E until now.

The identifiers of the OBJECTs are written totally lowercase!!

Copy this file to "EMODULES:libraries/" and to use it write in your source:

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

See also:

```

Modules/libraries/muip.m
and
Modules/libraries/Ev3_mui.m
.
```

## 1.9 Files: Modules/libraries/Ev3\_mui.m

```

"Modules/libraries/Ev3_mui.m"
~~~~~

```

This is a special version of  
 mui.m  
 for the new AmigaE v3. There are two  
 differences in this file:

- The OBJECTs are using the new features ("PTR TO xyz" is defined as "PTR TO xyz" and not as "LONG" etc.)
- The \_\_dummyxfc2\_\_ OBJECT is included (see MacroFiles/Ev3\_mui.ma).

If you have the new AmigaE v3, and if you want to use this file, then copy it to "EMODULES:libraries/" and rename it to "mui.m". The extra

---



information in this module causes no problems with older sources, I think.

## 1.10 Files: Modules/libraries/muip.m

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

This e-module 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 hadn't included them in former releases. But Lionel Vintenat (Mac2E) told me, that one reason for including his own MUI-files with Mac2E v4 was my incomplete

```
mui.m
file.
```

OK, now ALL struct's are available as OBJECT's, but not all in one file! The (I think) very useless MUIP\_... struct's are in this extra module. The

```
mui.m
module is big enough, even without this OBJECT's! If you really
need them, write
```

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

in your source. (And please send me

a mail to tell me the reason! I know absolutly no situation where this OBJECT's are usefull in E!!)

See also:

```
Modules/libraries/Ev3_muip.m
```

## 1.11 Files: Modules/libraries/Ev3\_muip.m

```
"Modules/libraries/Ev3_muip.m"
~~~~~
```

This is a special version of  
muip.m  
for the new AmigaE v3. There is  
only one difference:

- The OBJECTs are using the new features ("PTR TO xyz" is defined as "PTR TO xyz" and not as "LONG" etc.)

If you have the new AmigaE v3, and if you want to use this file, then copy it to "EMODULES:libraries/" and rename it to "muip.m".

---

## 1.12 Files: MacroFiles/mui.ma

```

"MacroFiles/mui.ma"
~~~~~

```

This file contains the most macro definitions of the C mui.h file which are not simple CONSTs and because of that not in the

```

    mui.m
    module. Only
"the most", because some of them couldn't be translated to E v2.1b, but they
are not very important. (See
    MacroFiles/Ev3_mui.ma
    if you have E v3.)

```

To use this file and its macrodefinitions you need the program  
Mac2E

and for more informations about the macros, have a look at the comments of the mui.h file that came with MUI.

As a result of moving the CONSTs to the E-module-file mui.m, instead of using only one macrofile like the one that came with Mac2E up to version 4, it's possible to write MUI programs \*without\* using this macrofile and Mac2E, but I wouldn't do it!!! With this macrofile it's possible to write MUI programs in E nearly the same way as in C.

The more interesting advantage of putting the CONSTs in a E-module is the much more readable source after using Mac2E. Sometimes - not often, but sometimes - it's better to search an error in the source that was created by Mac2E and than it's better to read "MUIA\_String\_Contents" instead of "\$80428FFD".

The macronames are written the same way as the C originals in mui.h - with one exception: The macro "String()" is renamed to "StringMUI()" to avoid conflicts with the E-Function String()

Copy this file to ... where you want it - for example in a MacroFiles-directory in your Amiga-E-directory.

See also:

```

    MacroFiles/mui.pma
    and
    MacroFiles/Ev3_mui.ma
.

```

## 1.13 Files: MacroFiles/Ev3\_mui.ma

```

"MacroFiles/Ev3_mui.ma"
~~~~~

```

This is a special version of  
mui.ma  
for the new AmigaE v3. There is  
only one difference:

---

- The macros who are using the `__dummyxfc2__` struct (see: `Ev3_mui.m`) are only possible with AmigaE v3 and not included in the macrofile `"mui.ma"` for Ev2.1b user.

## 1.14 Files: MacroFiles/mui.pma

```

"MacroFiles/mui.pma"
~~~~~

```

This is the pre-analysed version of the macrofile  
 MacroFiles/mui.ma  
 for direct use with  
 Mac2E  
 v4.0.

## 1.15 Files: MacroFiles/Ev3\_mui.pma

```

"MacroFiles/Ev3_mui.pma"
~~~~~

```

This is the pre-analysed version of the macrofile  
 MacroFiles/Ev3\_mui.ma  
 for direct use with  
 Mac2E  
 v4.0.

## 1.16 Files: Sources/doMethod.e

```

"Sources/doMethod.e"
~~~~~

```

To write MUI programs, you need a DoMethod function. Because this function is part of the `amiga.lib`, which we don't have as E programmer, you have to use your own one, or better use this one written by Wouter van Oortmerssen.

Add it into your source to use it and have a look at the demo-source

```

Sources/MUI_Demo.em
to see, how to use it.

```

If you use AmigaE v3, you can find this function in the emodule `"EMODULES:tools/Boopsi.m"` which comes with Ev3, too.

---

## 1.17 Files: Sources/installhook.e

```
"Sources/installhook.e"
~~~~~
```

Sometimes - for example if you use PopUps - you need hooks. This source gives you a PROC that makes it very easy to install the needed hook-structures. Look at this source (the comments) and the demo-source

```
Sources/PopUp.em
to see, how to use it.
```

This source (without the comments) and an E-module of it can also be found in the AmigaE v3 distribution. It's only still included for users of AmigaE v2.1b.

NOTE: Normally a hookfunktion has three arguments. But the first one, a ptr to the hookstructure, is (in most cases) not very interesting. I'd never needed this pointer and so I changed the installhook-source. This changed source was included in the mui22Edev.lha as installhook2. But now I noticed that it's possible to use the original installhook-source with only two argumented hookfunktionen, too. Why? Installhook uses an assembler-interface to call the hookfunktion. One thing this interface does, is putting the 3 arguments on the stack \*and\*remove\*them\* after the hookfunktion-call. It's no problem if a hookfunktion like "PROC hf(m,o)" only uses the last two arguments on the stack (without the ptr to the hookstructure) and not all three. (Have a look at the comments in installhook.e)

## 1.18 Files: Sources/MUI-Demo.em

```
"Sources/MUI-Demo.em"
~~~~~
```

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. This source contains macros:

```
Mac2E
is needed
to build the file
MUI-Demo.e
for the E-compiler.
```

See also:

```
Problems
```

## 1.19 Files: Sources/MUI-Demo.e

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

This is the result after replacing all macros in  
 MUI-Demo.em  
 with the  
 macrofile  
 mui.pma  
 and Mac2E.

## 1.20 Files: Sources/PopUp.em

"Sources/PopUp.em"  
 ~~~~~

This is a translation of the demo-source "PopUp.c" that came with MUI2.1.  
 Use it to see, how to use the  
 installhook  
 function.

## 1.21 Files: Sources/PopUp.e

"Sources/PopUp.e"  
 ~~~~~

This is the result after replacing all macros in  
 PopUp.em  
 with the  
 macrofile  
 mui.pma  
 and Mac2E.

## 1.22 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?

---

## 1.23 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.m  
. Use this one instead of TRUE whenever you want  
to give TRUE to MUI. See "Sources/MUI\_Demo.em" and "Sources/PopUp.em".

## 1.24 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 then MUI 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 than 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 nnsset()) in the files

MacroFiles/mui.ma  
and  
MacroFiles/Ev3\_mui.ma  
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 nnsset() of former releases is now obsolete.

INFO: You should use THIS set() and not the one that came with Mac2E v4, because that one can cause problems if you write something like:

```
set( func_getobj(), MUIA_xyz, value ) or
set( obj_array[]++, MUIA_xyz, value ) or
set( x:=func(), MUIA_xyz, value )
```

## 1.25 Problems: TAG\_IGNORE,0

```

                                TAG_IGNORE,0
~~~~~
```

Some of the macros in  
MacroFiles/mui.ma  
and  
MacroFiles/Ev3\_mui.ma  
are

ending with "[TAG\_IGNORE,0,". That seems to be superfluous but it is needed, because in Ev2.1b if you want to split a statement over several lines, you can't end a line with "[" but with a comma. And in "Ev3\_mui.ma" it's needed to allow the same use of the macros. (See program "OptiMUI" that comes with "Mac2E".)

## 1.26 Copyrights and authors

```

                                COPYRIGHTS AND AUTHORS
~~~~~
```

Jan Hendrik Schulz (that's me): - this ReadMe guide  
(

```

    my address
    ) - all files in Modules/ and MacroFiles/
    (created from the original C mui.h file)
    - translations of MUI-Demo and PopUp to E
```

Stefan Stuntz: - MUI with mui.h and the original C sources  
of MUI-Demo.em and PopUp.em

Lionel Vintenat: - Mac2E

Wouter van Oortmerssen: - AmigaE  
- doMethod.e

---

- installhook.e (comments added by me)

## 1.27 My address

MY ADDRESS  
~~~~~

Snail mail: Jan Hendrik Schulz  
Reinhard-Raffalt-Str. 41  
94036 Passau  
Germany

Internet: schulzj@fmi.uni-passau.de

## 1.28 Future

FUTURE  
~~~~~

With future versions of  
MUI  
this files will be included. (If Stefan  
don't forget it again like now with MUI 2.2)

If you have some ideas to make things better, please inform  
me  
about  
them!

## 1.29 Bugs

BUGS  
~~~~~

The PupUp-example hangs if you try to close it while there are still some  
ASL-PopUps open. I don't know why! If anybody sees the bug, please send  
me  
a mail.

If you find a bug in the macrofiles, the modules or this textfile, please  
send  
me  
a mail to give me the chance to remove it.

---



## 1.30 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!)