

**AddComment**

**COLLABORATORS**

	<i>TITLE :</i> AddComment		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 2, 2023	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>AddComment</b>	<b>1</b>
1.1	Overview to AddComment.guide . . . . .	1
1.2	Overview . . . . .	1
1.3	System Requirements . . . . .	2
1.4	Installation . . . . .	2
1.5	Usage . . . . .	2
1.6	The surface . . . . .	3
1.7	The menus . . . . .	4
1.8	The ToolTypes . . . . .	4
1.9	The ARexx commands . . . . .	5
1.10	The configuration file . . . . .	5
1.11	A sample configuration file . . . . .	6
1.12	Integration Tips . . . . .	8
1.13	History of AddComment . . . . .	8
1.14	To do . . . . .	8
1.15	Copyrights . . . . .	9
1.16	Sample for a Program Head . . . . .	9
1.17	Sample for a Function Head . . . . .	10
1.18	Sample for a Description . . . . .	10

## Chapter 1

# AddComment

### 1.1 Overview to AddComment.guide

Contents of this documentation

Overview

System Requirements

Installation

Usage

Integration Hints

History

ToDo

Copyrights

Copyright by Harald Sodemann 1996-97

### 1.2 Overview

AddComment is a small Commodity that makes it easy to insert ↔  
comment boxes  
into the source loaded in an editor.

It is designed to be  
integrated  
in an development system,  
therefore it is a Commodity and it has an  
ARexx-Host  
.

It offers three forms, for

---

Programm Heads

,

Function Heads

and

Description boxes

.

Standart texts like author, date, version and longer descriptions can be entered and formatted as left, center and right.

The communication between AddComment and the editors is done by ARexx, which makes it necessary that the controlled editor has an ARexx-Host, too.

See also

integration tips

!

### 1.3 System Requirements

AddComment requires besides OS2.0 a fitting development environment, which means an arexx-speaking editor, if possible with configurable menus.

Also, ARexx has to be installed and the RexxSysLib must be in Libs:

AddComment has been tested on a A500 with 1MB Chip and 4MB Fast, but there should not be any problems on other Amigas since AddComment uses very little memory.

### 1.4 Installation

Simply drag AddComment into a drawer, if you want to have it ←

always running

put it in your SYS:WBStartup drawer.

You also have to write a

configuration file

.

Compatibility tests

AddComment was tested with the following editors:

- MaxonDev 4.0
- GoldEd 3.0
- Edward 1.0
- OEd
- TurboText
- Ed 2.0.

### 1.5 Usage

---

The issues:

The surface

The menus

The ToolTypes

The ARexx commands

The configuration file

## 1.6 The surface

The control gadgets

Form: Toggles between the three forms.  
 New: Deletes all string gadgets und lists.  
 Insert: Sends the current comment box to the editor.  
 Insert&Close: Sends the comment box and closes the window afterwards.  
 Cancel: Closes the window.

The form program head

```

sample box
    Programname:  Titel of the commentbox
Author:  Author of the sourcecode
Version:  Version of the sourcecode
Language:  Used programming language
Description:  List for descriptions etc.
Add:  Add a new line to the list
Del.:  Delete marked line
Cyclegadget:  Select the setting of the marked line, toggle between
                L for left, C for center and R for right orientation.
  
```

The form function head

```

sample box
    Function name:  Name of the function
List for:  Toggle between parameter and description list
Add:  see above
Del.:  see above
Cyclegadget:  see above
  
```

The form description

```

sample box
    Add:  see above
Del.:  see above
Up:  Move marked line one up
Down:  Move marked line one down
Cyclegadget:  see above
  
```

Key controlling

---

Almost all gadgets have key shortcuts, remarkable at the underlined characters. The cyclegadgets for the orientation of the list lines can be toggled by pressing the space bar.

In the form programhead you may jump through the stringgadgets via TAB.

The stringgadgets accept only as many characters as selected via the tooltype BOXWIDTH, this prevents lines from being cutted at the end.

## 1.7 The menus

The menu Project

About...	Shows a short info text and the current hotkey
Hide	Closes down the window
Quit	Really quits AddComment

The menu Language

Consists of up to 10 languages defined in the config file. One can be selected and the data is used for the build up of the boxes.

The menu Editor

Consists of up to 10 editors defined in the config file. After the selection of a editor the window closes and opens again to eventually open up on a PubScreen. By the way, if a PubScreen cannot be found, the window opens on the default PubScreen.

## 1.8 The ToolTypes

The program accepts the following ToolTypes:

CX_PRIORITY:	sets the priority of the Commodity in the input stream, default is 0.
CX_POPUP:	YES for opening the window at the start, NO for not. default is YES.
CX_POPKEY:	valid eventstring for the hotkey, default is ralt lalt a.
CONFIGFILE:	DOS-Name to the configfile, default is PROGDIR:addcomment.config
AUTHOR:	name of the author, no default value.
BOXWIDTH:	width of the comment boxes including comment sequences, default is 60, good are values between 45 and 75.
FORM:	selects the form presented at popup, default is 0 (from 0-2).
EDITOR:	selects the editor from the configfile counting from 0, default is 0.
LANGUAGE:	selects the language, default is 0.
LEFTEDGE:	left edge of the window at popup in pixels, default is 80
TOPEDGE:	top edge of the window at popup in pixels, default is 40

## 1.9 The ARexx commands

The program can be controlled via ARexx in the most essential things. The Hosts name ist "ADDCOMMENT", below the list of commands:

command, abbrev.	description
popup, pu	Opens AddComment's window or brings it to the front.
popdown, pd	Closes AddComment's window
insert, in	Sends the currently active box data to the editor
closeinsert, ci	Sends the box data and closes the window afterwards.
programform, pf	Selects the programhead form
functionform, ff	Selects the functionhead form
descriptionform, df	Selects the description form
die	Removes AddComment from memory

You may call any command at any state of the window, it doesn't matter whether it's open or closed, this causes no problems.

## 1.10 The configuration file

AddComment needs a configuration file to be adjusted to your editors and programming languages, here is the description of the file format: ↵

Data for the editors

Each editor, that shall be listed in the menu Editor needs an entry in the configfile like this:

<Editor>	editor's name, appearing in the menu, e.g. Eddie
<PublicScreen>	name of the editors PublicScreen
<ARexxPort>	name of editor's ARexx host
<Quote Mark>	Character, with that text send with ARexx will be enclosed Necessary for Ed and GoldEd, use " for them. Most other editors don't require the quote marks
<Lock>	command for locking the editor before sending arexx commands to it, used by GoldEd
<UnLock>	command for unlocking the editor again, used by GoldEd
<Return>	ARexx command of the editor for "RETURN"
<Insert>	command for inserting characters
<Line start>	command for jumping to the first character in the line
<Text start>	command for jumping to the top of the text

AddComment uses the commands in the following order:

When inserting a program head, AddComment sends the commands

<Text start><Line start>



At each line of text the commands

```
<Insert><Return>
```

are sended.

The text insert command line is built like

```
' <Insert Command> <Quote Mark> <text to be inserted> <Quote Mark> '.
```

IMPORTANT: For not used commands, be sure to leave a blank line!!

Data for the languages

Each language, that shall be listed in the menu "Language" needs an entry in the configfile like this:

```
<Language>          language name, appearing in the menu
<start comment>    Characters at the beginning of a comment, e.g. "/*" in C
<end comment>      Characters at the end of a comment if existing.
<frame character>  Character to build the boxes frame of.
<bar character>    Character to build separation lines of.
```

IMPORTANT: GoldEd has problems on using "\*" as frame character, as it uses this character already for other things. So you have to use an other frame character, e.g. ##, +, & etc.

The characters for end comment are not necessarily needed, but if there is none, enter blank line instead.

See also the

```
sample configuration file
```

```
.
```

## 1.11 A sample configuration file

This is a typical configuration file with two editors and languages defined. It is possible to enter comments to the configuration file, therefore the first character of the line must be '#', and the comment continues to the end of the line.

If you want to use '#' as a character for e.g. a frame character, you have to double it. (looks like '##')

The data has to occur in a special order, first, you write the number of editor descriptions, then come the editor descriptions, then comes the number of language descriptions and then the language descriptions themselves. If a argument like a ARexx command is not supported, write a linefeed or blank line instead.

```
# Configfile for AddComment V1.2
# © by Harald Sodemann 30.8.1997
# Number of editors
2
# name of the first editor
MaxonDev4.0
# name of the PublicScreen
MaxonDevelop.1
# name of the ARexxPort
MAXONDEV.1
# quote mark, not required -> blank line
```

```
# lock command, not required

# unlock command, not required

# return command
'text_return'
# insert text command
'text_insertchars '
# line start command
'text_linestart'
# text start command
'text_textstart'
# name of the second editor
GoldEd3.0
Workbench
GOLDED.1
# quote mark, very important for GoldEd
"
# lock command, very important for GoldEd
'LOCK CURRENT'
# unlock command, very important for GoldEd
'UNLOCK'
# return command, not required by GoldEd

# insert and return command
TEXT CR T=
# line start
'GOTO COLUMN 1'
# text start
'GOTO TOP'
# end of the editor descriptions
# number of languages
2
# name of the first language
C++ 2.0
# start comment
//
# end comment, not required

# character for box frames, * will not be accepted by GoldEd!!
# use e.g. ##, +, & etc. instead!
*
# character for separation bars
-
#name of the second language
Pascal
# start comment
{*
# end comment
*}
# frame character
*
# bar character
-
#
# End of sample configuration file.
```

AddComment starts also if it cannot find a config file, then it uses ANSI C as language and Ed as editor.

## 1.12 Integration Tips

AddComment was designed to be easily integrated in a developing environment. There are many possibilities for a good integration, here are some hints:

Make the functions of AddComment accessible by menu and hotkeys!  
Use ARexx-Scripts like this:

```
/* Open AddComment with program form */
address "ADDCOMMENT"

'pf' /* choose program form */
'pu' /* open addcomment window */
```

If your Editor has configurable menus, enter following menu items:

```
MenuItem

"Program Box"
"Function Box"
"Description "
```

Also put these command sequences on function keys, e.g. F8-F10.

Also use the ToolType CX\_POPKEY for a harmonic integration of AddComment to your developing environment.

## 1.13 History of AddComment

```
*****
*                               History of AddComment by Harald Sodemann                               *
*-----*
* Version 0.1:   CLI-Output, just for alrorphism testing           (1.4.96)*
*-----*
* Version 1.0D/E: First working release, in German and English (1.9.96)*
*-----*
* Version 1.2D/E: Fixed some bugs:                               (1.9.97)*
* - ARexx-Commands aren't sent to the editors directly any more but to *
*   the rexxmast-program.                                           *
* - Included new configfile-parameters for compatibility with GoldEd *
*   and Ed.                                                         *
*****
```

## 1.14 To do

I still have some good ideas for new features:

- \* An interactive mode, that steps through the program text, popping up at every function with the prepared form, only requesting a description for the parameters and the functions purpose.
- \* A possibility to change the commentbox width online.
- \* More ARexx commands for putting text in gadgets, selecting the editor etc.
- \* Localisation of AddComment under OS2.1
- \* Sizeable Window with dynamic gadget adjust

I'd like to hear more new ideas or improvements from you, this would give me some motivation to continue the development of AddComment.

See my

Address  
!

## 1.15 Copyrights

AddComment is freeware, use it as often as you like, enter anything in the gadgets you like, just enjoy it.

Originally, AddComment is only a by-product of a C++-ClassLibrary I'm developing for the Amiga, with easy Window-, Gadget-, Port-, Errorhandling and a lot of more fascinating things.

So AddComment is something like a "guinea pig" of this ClassLibrary, but it works quite fine and is sometimes useful.

If you think so, too or if you are curious about my ClassLibrary or find some bugs or what-do-I-know-what, write me!

harald.sodemann@stud.uni-bayreuth.de

- End of the documentation -

## 1.16 Sample for a Program Head

This is a sample program head, created with AddComment:

```
//*****
//*                               TestProgramm.cpp                               *
//*-----*
//* Author:Harald Sodemann      Version   23.789 *
//* Date   :31.Februar 1998      Language:C++ 2.0 *
//*-----*
//* Here could stand something useful, if I had *
//* the head to do that so late in the night! *
//*****
```

## 1.17 Sample for a Function Head

This is a sample function head, created with AddComment:

```
//*****  
//*      int TestFunction(int Testx, int Testy)  *  
//*-----*  
//* Testx:  is not so important!                *  
//* Testy:  is even less important.            *  
//*                                               *  
//* return value: Simply forget it!.          *  
//*-----*  
//*      This function has only testing purposes. *  
//*****
```

## 1.18 Sample for a Description

This is a sample description box, created with AddComment:

```
//*****  
//*      Some demonstration of the              *  
//*      formatting possibilities:              *  
//*                                               *  
//* left position                               *  
//*      center position                       *  
//*      right position                        *  
//*                                               *  
//*      Great, isn't it?!                     *  
//*****
```