

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

| | | |
|----------|--|----------|
| 1 | AddComment | 1 |
| 1.1 | Übersicht zu AddComment.guide | 1 |
| 1.2 | Kurzbeschreibung | 1 |
| 1.3 | Systemvoraussetzungen | 2 |
| 1.4 | Installation | 2 |
| 1.5 | Bedienung des Programms | 3 |
| 1.6 | Die Bedienoberfläche | 3 |
| 1.7 | Die Menüs | 4 |
| 1.8 | Die ToolTypes | 4 |
| 1.9 | Die ARexx-Befehle | 5 |
| 1.10 | Die Konfigurationsdatei | 5 |
| 1.11 | Eine Beispielkonfigurationsdatei | 6 |
| 1.12 | Integrationshinweise | 8 |
| 1.13 | Versionsüberblick | 9 |
| 1.14 | zukünftige Erweiterungen | 9 |
| 1.15 | Copyrighthinweise | 9 |
| 1.16 | Beispiel für einen Programmkopf | 10 |
| 1.17 | Beispiel für einen Funktionskopf | 10 |
| 1.18 | Beispiel für einen Beschreibungstext | 10 |

Chapter 1

AddComment

1.1 Übersicht zu AddComment.guide

Inhalt dieser Dokumentation

Kurzbeschreibung

Systemvoraussetzungen

Installation

Bedienung

Integrationsvorschläge

Versionsübersicht

zukünftige Erweiterungen

Copyrightvermerke

Copyright Harald Sodemann 1996-97

1.2 Kurzbeschreibung

AddComment ist ein kleines Commodity, daß es auf einfache Weise ↔
ermöglicht,
Kommentarboxen in den in einem Editor geladenen Quellcode einzufügen.

Es ist dazu gedacht, in eine Entwicklungsumgebung
integriert
zu
werden, deshalb ist es als Commodity ausgelegt und besitzt eine

ARexx-Schnittstelle

.

Es bietet drei Formulare, für

 Programmbeschreibungen

,

 Funktionsbeschreibungen

 und

 allgemeine Hinweistexte

.

Dabei können Standarttexte wie Autor, Version etc. genauso wie längere Texte eingegeben werden, die dann rechtsbündig, linksbündig oder zentriert werden können.

Die Kommunikation zwischen AddComment und dem jeweiligen Editor erfolgt dabei mit ARexx, d.h. wenn AddComment einen Editor ansteuern soll, muß dieser über einen ARexx-Host verfügen.

Siehe auch

 Integrationsempfehlungen

!

1.3 Systemvoraussetzungen

AddComment benötigt außer OS2.0 nur eine passende Entwicklungsumgebung, d.h. mit einem arexxfähigen Editor, am besten mit konfigurierbaren Menüs. Außerdem muß ARexx installiert sein, ebenso wird die REXXSYSLIB benötigt. Getestet wurde es zwar nur auf einem A500 mit 1MB ChipRAM und 4MB FastrAM, aber es sollte eigentlich auf allen Amigas funktionieren :-).

1.4 Installation

AddComment einfach in eine Schublade ziehen, in der es Platz hat. Es empfiehlt sich allerdings ganz besonders der Ordner WBStartup, wenn man gleich loskommentieren will.

Zudem muß eine

 Konfigurationsdatei
 geschrieben werden.

Kompatibilität

AddComment wurde mit verschiedenen Editoren ausprobiert, und zwar

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

Mit all diesen Editoren arbeitet AddComment klaglos zusammen.

Die meisten Editoren benötigen allerdings eine ihren Besonderheiten angepaßte

Konfigurationsdatei

1.5 Bedienung des Programms

Die einzelnen Bereiche:

Die Bedienoberfläche

Die Menüs

Die ToolTypes

Die ARexx-Befehle

Die Konfigurationsdatei

1.6 Die Bedienoberfläche

Die Steuerungsgadgets

Formular: Blättert durch die drei Formularblätter.
 Neu: Löscht alle Stringgadgets und Listen.
 Einfügen: Sendet die gewählte Kommentarbox an den Editor.
 Einf.& Zu: Sendet die Kommentarbox und schließt danach das Fenster.
 Abbruch: Schließt das Fenster.

Das Formular Programm

Beispielbox

Programmname: Titel der Kommentarbox

Autor: Autor des Programms
 Version: Version des Programms
 Sprache: Verwendete Sprache
 Beschreibung: Liste für Erklärungen etc.
 Hinz.: Neue Zeile zur Liste hinzufügen
 Lösch.: Markierte Zeile löschen
 Cyclegadget: Ausrichtung der gewählten Zeile einstellen,
 L für linksbündig, Z für zentriert, R für rechtsbündig.

Das Formular Funktion

Beispielbox

Funktionsname: Name der Funktion

Liste: Umschalten zwischen Liste für Argumente und Beschreibungen
 Hinz.: wie oben
 Lösch.: wie oben
 Cyclegadget: wie oben

Das Formular Beschreibung

Beispielbox

Hinz.: wie oben
 Lösch.: wie oben

Hoch: Markierte Zeile vor die vorhergehende Zeile bringen
Runter: Markierte Zeile nach die nachfolgende Zeile bringen
Cyclegadget: wie oben

Tastatursteuerung

Nahezu alle Gadgets sind per Tastendruck erreichbar, erkennbar an dem jeweils unterstrichenen Buchstaben.

Die Cyclegadgets zur Ausrichtung der Listenzeilen können über einen Druck der Leertaste weitergeschaltet werden.

Im Formular zur Programmbeschreibung kann man über TAB durch die Textgadgets springen.

Die Textgadgets akzeptieren nur Eingaben bis zur maximalen Länge, die über das ToolType BOXWIDTH eingestellt wurde, um zu verhindern, daß Text abgeschnitten wird.

1.7 Die Menüs

Das Menü Projekt

Über... Zeigt einen kleinen Hinweistext und den aktuellen Hotkey an
Verstecken Schließt das Fenster von AddComment
Beenden Entfernt AddComment aus dem Speicher

Das Menü Sprache

Beinhaltet bis zu 10 Programmiersprachen, abhängig von der Konfigdatei. Es kann jeweils nur eine ausgewählt sein; die zugehörigen Daten erscheinen dann in den Gadgets, z.B. beim Gadget Sprache im Formular zur Programmbeschreibung.

Das Menü Editor

Beinhaltet bis zu 10 Editoren, abhängig von der Konfigdatei. Nach der Auswahl eines Editors wird das Fenster geschlossen und mit den zugehörigen Daten geöffnet, besonders auch auf einem eventuell anderen PubScreen. Kann der PubScreen nicht gefunden werden, öffnet das Fenster auf dem DefaultPublicScreen.

1.8 Die ToolTypes

Das Programm versteht folgende ToolTypes

CX_PRIORITY: Stellt die Priorität des Commodities im Inputstream ein, voreingestellt ist 0.
CX_POPUP: YES wenn das Fenster beim Starten gleich geöffnet werden soll, NO wenn nicht, voreingestellt ist YES.
CX_POPKEY: gültiger String für den Hotkey, voreingestellt ist ralt lalt a.
CONFIGFILE: Der gültige DOS-Name mit Pfad zur Konfigurationsdatei, voreingestellt ist PROGDIR:AddComment.config.

AUTHOR: Der Name des Programmators für das erste Formular, keine Voreinstellung.

BOXWIDTH: Die Breite der Kommentarbox in Zeichen inklusive evtl. Kommentarzeichen, voreingestellt ist 60, empfehlenswert sind Breiten zwischen 45 und 70.

FORM: Das beim Öffnen des Fensters angezeigte Formular kann hier eingestellt werden, Werte von 0,1,2 sind zulässig, entsprechend den Formularen Programm, Funktion und Beschreibung. Voreingestellt ist Formular 0.

EDITOR: Der im Menü Editor gewählte Editor kann hier vorgewählt werden, die Werte können von 0 bis zur in der Konfigdatei definierten Anzahl von Editoren-1 gehen, voreingestellt ist 0.

LANGUAGE: Hier kann die Sprache vorgewählt werden, (wie bei den Editoren), voreingestellt ist 0.

LEFTEDGE: Die linke Ecke des Fensters beim Öffnen in Pixels

TOPEdge: Die obere Ecke des Fensters beim Öffnen in Pixels

1.9 Die ARexx-Befehle

Das Programm kann in den wichtigsten Bereichen über ARexx gesteuert werden, der Hostname ist "ADDCOMMENT", die Befehle sind:

| Befehl, Abkürzung | Funktion |
|---------------------|---|
| popup, pu | Öffnet das Fenster von AddComment oder bringt es in den Vordergrund. |
| popdown, pd | Schließt das Fenster von AddComment |
| insert, in | Sendet die Daten des gewählten Formulars an den Editor, funktioniert auch bei geschlossenem Fenster |
| closeinsert, ci | Sendet die Daten des gewählten Formulars and den Editor und schließt danach das Fenster. |
| programform, pf | Wählt das Programmformular aus, auch bei geschlossenem Fenster |
| functionform, ff | Wählt das Funktionsformular aus |
| descriptionform, df | Wählt das Beschreibungsformular aus |
| die | Beendet das Programm AddComment |

Es ist kein Problem für AddComment, auch bei geschlossenem Fenster das Formular zu wechseln oder Boxen zu erzeugen und zu senden.

1.10 Die Konfigurationsdatei

Um AddComment auf seine eigenen Editoren und Programmiersprachen einzu-
stellen, muß man ein Konfigurationsfile erstellen, das einen bestimmten
Aufbau hat und zu den jeweiligen Editoren/Sprachen bestimmte Daten liefert.

Die Daten zum Editor

Zu jedem Editor, der im Menü Editor aufgeführt werden soll, muß folgender Eintrag ins Konfigfile:

```

<Editor>           Name des Editors im Menü, z.B. Eddie
<PublicScreen>    Name des PublicScreens des Editors, wenn keiner
                   vorhanden ist empfiehlt sich der Eintrag "Workbench"
<ARexxPort>       Name des ARexx-Hosts des Editors
<Anführungszeichen>
                   Zeichen, in das der Text im ARexx-Befehl eingeschlossen
                   werden muß, erforderlich z.B. bei Ed und GoldEd;
                   Hier wird dann " benötigt. Bei den meisten anderen
                   Editoren ist das Anführungszeichen nicht erforderlich.
<Lock>            ARexxbefehl zum Verriegeln des Editors vor der
                   Ausführung von ARexxbeehlen; erforderlich bei GoldEd.
<Unlock>          ARexxbefehl zum Entriegeln des Editors nach Ausführung
                   von ARexxbefehlen; erforderlich bei GoldEd.
<Return>          ARexxbefehl des Editors für "return"
<Einfügen>        ARexxbefehl für Text einfügen
<Zeilenanfang>    ARexxbefehl für Zeilenanfang anspringen
<Textanfang>      ARexxbefehl für Textanfang anspringen

```

Die Befehle werden folgendermaßen verwendet:

Beim Einfügen eines Programmkopfes wird zunächst die Folge

<Textanfang><Zeilenanfang> gesendet.

Bei jeder Zeile weiteren Textes wird dann die Folge

<Einfügen><Return> gesendet.

Die Textzeilen selbst bestehen aus

' <Einfügen> <Anführungszeichen> Text <Anführungszeichen> '.

Für nicht benötigte Befehle muß im Konfigfile eine Leerzeile stehen!

Die Daten zur Sprache

Zu jeder Sprache, die im Menü aufgeführt werden soll, muß folgender Eintrag ins Konfigfile:

```

<Sprache>         Name der Sprache im Menü
<Kommentarbeginn> Zeichenfolge für Kommentarbeginn, z.B. "/*" für C
<Kommentarende>   Zeichenfolge für Kommentarende wenn vorhanden
<Rahmenzeichen>   Zeichen für den Rahmen der Box.
<Linienzeichen>   Zeichen, aus dem Trennlinien aufgebaut werden sollen.

```

WICHTIG: GoldEd kommt mit dem Zeichen "*" als Rahmensymbol nicht zurecht, da es bereits eine andere Funktion besitzt. Stattdessen muß ein anderes Symbol in der Konfigdatei angegeben werden, z.B. ##, +, & etc.

Die Zeichenfolge für Kommentarende muß nicht unbedingt angegeben werden, es hat ja auch nicht jede Sprache dafür ein Zeichen (wie z.B. Assembler).

Siehe dazu die

Beispielkonfigurationsdatei

1.11 Eine Beispielkonfigurationsdatei

Hier ein typisches Konfigurationsfile mit je zwei Editoren und Sprachen. Man kann in das Konfigfile auch Kommentare einfügen, dazu muß das erste Zeichen einer Zeile '#' sein. Kommentare sind immer ganzzeilig. Soll '#' als erstes Zeichen in einer Zeile stehen und kein Kommentar sein, z.B. wenn man eine Box aus '#' aufgebaut haben will, so schreibt man '##'. Die Reihenfolge der Daten ist festgelegt: Zuerst kommt die Anzahl der Editoren als Zahl, dann die Datensätze zu den Editoren, dann die Anzahl der Sprachen als Zahl, zuletzt die Datensätze zu den Sprachen. Wenn ein Argument nicht angegeben wird, wie z.B. das Kommentarende, so muß stattdessen ein Linefeed stehen.

```
# Konfigurationsdatei zu AddComment V1.2
# © by Harald Sodemann 30.8.1997
# Anzahl der Editoren
2
# Name des ersten Editors
MaxonDev4.0
# Name des Public Screen
MaxonDevelop.1
# Name des ARexx Ports
MAXONDEV.1
# Anführungszeichen, nicht benötigt -> Leerzeile

# Lock-Befehl, nicht benötigt

# UnLock-Befehl, nicht benötigt

# Return-Befehl
'text_return'
# Text einfügen-Befehl
'text_insertchars '
# Zeilenanfang-Befehl
'text_linestart'
#Textanfang-Befehl
'text_textstart'
# Name des zweiten Editors
GoldEd3.0
Workbench
GOLDED.1
# Anführungszeichen, unbedingt notwendig bei GoldEd
"
# Lock-Befehl, unbedingt notwendig bei GoldEd
'LOCK CURRENT'
# UnLock-Befehl, um Blockade von GoldEd aufzuheben
'UNLOCK'
# Return-Befehl, bei GoldEd nicht notwendig

# Einfügen-Befehl mit integriertem Return-Befehl
TEXT CR T=
# Zeilenanfang
'GOTO COLUMN 1'
# Textanfang
'GOTO TOP'
# Ende der Editorbeschreibungen
# Anzahl der Sprachen
```

```

2
# Name der ersten Sprache
C++ 2.0
# Kommentareinleitung
//
# Kommentarende; nicht benötigt

# Zeichen für Kommentarboxrahmen, * wird von GoldEd nicht akzeptiert!!!
# für GoldEd empfiehlt sich z.B. ##, +, & etc.
*
# Zeichen für Trennstriche
-
#Name der nächsten Sprache
Pascal
# Kommentaranfang
{*
# Kommentarende
*}
# Rahmenzeichen
*
# Trennstrichzeichen
-
# Ende der Konfigurationsdatei

```

Grundsätzlich läßt sich AddComment auch ohne Konfigurationsdatei starten, es sind dann die Sprache ANSI C und der Editor Ed fest eingestellt.

1.12 Integrationshinweise

AddComment wurde speziell auf die Integration in Entwicklungsumgebungen hin optimiert. Für eine gelungene Integration gibt es viele Möglichkeiten, hier nur ein paar Tips.

In Editoren mit konfigurierbaren Menüs kann man folgende Befehle aufnehmen:

MenuItem-Text ARexx-Script

```

"Programmbox "    rx script1 ;wählt Formular Programm und öffnet AddComment
"Funktionsbox"    rx script2 ;wählt Formular Funktion und öffnet AddComment
"Beschreibung"    rx script3 ;wählt Beschreibung und öffnet AddComment

```

z.B.

```
/* script1 */
```

```
address "ADDCOMMENT"
```

```
'pf' /* wähle Programmformular */
'pu' /* öffne AddComment-Fenster */

```

Diese ARexx-Scripts könnte man auch über Funktionstasten starten.

Durch den in den ToolTypes anpassbaren Hotkey läßt sich AddComment auch hier harmonisch in die Entwicklungsumgebung einfügen.

1.13 Versionsüberblick

```

*****
*           History of AddComment by Harald Sodemann           *
*-----*
* Version 0.1:   CLI-Output, just for alrorithm 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 zukünftige Erweiterungen

Ich hab noch `ne ganze Menge Ideen, was noch ins Programm reinpassen würde,

- * ein interaktiver Modus, in dem das Programm den Quelltext nach Funktionen durchforsten würde und dann bereits mit vorbereitetem Formular nur noch nach den Beschreibungen zu den Argumenten und der Erläuterung zur Funktion fragt.
- * Zudem wäre es nicht schlecht, wenn man die Boxenbreite während der Laufzeit einstellen könnte.
- * Ein paar ARexxbefehle mehr wären auch nicht schlecht.
- * Lokalisierung des Programms unter OS2.1
- * Anzeige des aktuellen Datums im Datumfeld
- * Fenster mit Size-Gadget und dynamischer Anpassung der Gadgets

Über zusätzliche Anregungen oder Anfragen, die den Sinn einer Weiterentwicklung bestätigen würden, würde ich mich freuen.

1.15 Copyrightinweise

AddComment ist Freeware, d.h. macht damit was ihr wollt sooft ihr wollt. Eigentlich ist AddComment ein Nebenprodukt einer C++-Klassenbibliothek, die ich gerade für den Amiga entwickle, mit Windowhandling, Gadgethandling, ARexxPorts und solchen tollen Sachen. Es ist sowas wie ein Versuchskaninchen, an dem die ersten Klassen sich austoben dürfen. Trotzdem ist es doch recht stabil und nützlich.

Wenn jemand einen Bug finden sollte oder Vorschläge für künftige Verbesserungen hat, sende er/sie bitte an mich unter:

harald.sodemann@stud.uni-bayreuth.de

- Ende der Dokumentation -

1.16 Beispiel für einen Programmkopf

Dies ist ein Beispiel für einen ausgefüllten Programmkopf, erstellt mit AddComment:

```
//*****  
//*          TestProgramm.cpp          *  
//*-----*  
//* Autor:Harald Sodemann      Version 23.789 *  
//* Datum:31.Februar 1998      Sprache:C++ 2.0 *  
//*-----*  
//* Hier könnte nun etwas einigermaßen wichtiges *  
//* stehen, wenn man dazu die Muße aufbringt.   *  
//*****
```

1.17 Beispiel für einen Funktionskopf

Dies ist ein Beispiel für einen ausgefüllten Funktionskopf, erstellt mit AddComment:

```
//*****  
//*      int TestFunktion(int Testx, int Testy) *  
//*-----*  
//* Testx:  Ist nicht besonders wichtig!      *  
//* Testy:  Ist auch nicht von Bedeutung.     *  
//*-----*  
//* Rückgabewert:  Kannste vergessen.         *  
//*-----*  
//* Diese Funktion dient nur zu Testzwecken.  *  
//*****
```

1.18 Beispiel für einen Beschreibungstext

Dies ist ein Beispiel für einen Beispielttext, erstellt mit AddComment:

```
//*****  
//*      Nun eine Demonstration der          *  
//*      Formatierungsmöglichkeiten:        *  
//*-----*  
//* Linksbündiger Text                      *  
//*      Zentrierter Text                    *  
//*      Rechtsbündiger Text                *  
//*-----*  
//*      Toll, stimmts?!                    *  
//*****
```