

Most

Version 1.56
23. April 1994

Most ist ein Textanzeiger ähnlich Less, More oder MuchMore. Most ist voll 2.0 kompatibel: Er unterstützt ARexx, die locale.library, XPK, Pipes, beliebig viele Fenster, unbegrenzt große Dateien und bietet Hilfe über den AmigaGuide. Most ist GiftWare.

von Uwe Röhm

Dokumentation zu Most V1.56

Datum: 23. April 1994

Autor: Uwe Röhm

Most ist ein Textanzeiger ähnlich Less, More oder MuchMore.

Most bietet

- beliebig viele Textfenster
- verteiltes Prozeßsystem
- Suchfunktion
- frei konfigurierbar
- Kopiermöglichkeit ins Clipboard
- Einladen von Dateien per Doppelklick
- Fontsensitive Oberfläche
- PubScreen, eigenen Bildschirm oder Workbench
- AppWindow, AppIcon
- ARexx Unterstützung
- Sprachanpassung per locale.library
- XPK Unterstützung
- Verwendung mit Pipes
- Hilfe per AmigaGuide
- benötigt Kickstart 2.0 oder höher!

1 Copyright

Am Anfang standen Schweiß und Mühe. Also sprach der Programmierer: Es werde Giftware! Und so war es.

Most 1.56 ist Giftware und darf frei kopiert und benutzt werden. Ich würde mich aber über Geschenke jeglicher Art von den Benutzern, die Most regelmäßig verwenden, **sehr** freuen.

Davon abgesehen behalte ich mir mein Copyright aber vor, also:

Copyright © 1992/93/94 Uwe Röhm

Ich verbiete ausdrücklich, daß Most kommerziellen Produkten ohne meine Zustimmung beigelegt wird. Insbesondere darf Most **nicht** für militärische Zwecke im weitesten Sinne verwendet werden. Im weitesten Sinne heißt, daß ich auch den Verwaltungen und sonstigen Büros solcher Firmen die Verwendung untersage, die militärische Produkte auch nur planen oder unterstützen (Hersteller sowieso).

Most darf in PD Sammlungen aufgenommen werden, solange jene nicht mehr als umgerechnet 10 DM kosten. Das ist das absolute Maximum! Ich verlange aber, daß Most immer vollständig, also insbesondere immer zusammen mit der Dokumentation verbreitet wird. Das gilt auch für jeden normalen Benutzer! Leute, denkt an die Doku - ohne die ist ein Programm nur die Hälfte wert!

2 Prinzip

Ich habe Most geschrieben, da ich eine Textanzeiger haben wollte, der auf der Workbench aufgeht und sowohl über die Tastatur als auch die Maus vollständig zu bedienen ist. Außerdem sollte er möglichst flexibel sein und insbesondere auch mit wenig Speicher auskommen.

Deshalb arbeitet Most **immer** mit einem Fenster, in dem der Text angezeigt wird. Wer will, kann jenes aber auch auf anderen Bildschirm öffnen lassen. Most kann sogar einen eigenen Bildschirm öffnen. Wieviele Fenster Most auf hat, wo jene liegen (welcher PubScreen) und welche Texte darin zu sehen sind hängt nur vom Speicher und dem Benutzer ab :-)

Jeder Text hat nicht nur sein eigenes Fenster, sondern dahinter steht auch ein eigener Prozeß. Damit sind die Textfenster voneinander völlig autonom. Insbesondere können ARexx Befehle an verschiedenen Texte gleichzeitig gesendet werden und es ist immer möglich, neue Fenster zu öffnen, auch wenn die bereits geöffneten gerade “beschäftigt” sind. Dazu gibt es einen Most Hauptprozess, der die Texte verwaltet und einige globale ARexx Befehle behandelt (see Chapter 11 [Der ARexx Port], page 23).

Um den Speicherverbrauch unter Kontrolle zu haben, arbeitet Most immer mit einem Puffer. Kleinere Texte, die vollständig in den Puffer passen, werden in einem Aufwasch eingelesen. Als Nebeneffekt wird die Textdatei nun nicht mehr benötigt und kann z.B. sofort weiter verändert werden. Zu große Texte werden pufferweise gelesen. Das spart Speicherplatz (man wähle kleine Puffergrößen), bringt aber weniger Komfort (sprich Zeiteinbuße) beim Scrollen und Section 5.3 [Suchen], page 17. Außerdem blockiert Most nun den Text, so daß er nicht verändert werden kann, solange Most in anzeigt.

Dieses “Paging” großer Dateien ist sehr nützlich, wenn man wenig Speicher oder übergroße Dateien hat. Wer es nicht haben will braucht nur die Puffer- größe sehr sehr groß machen (sagen wir mal 4 MB), womit wohl alle normalen Textdateien auf einen Schwung einlesbar sind. Keine Angst, Most verwendet dann natürlich nur soviel Speicher, wie die Datei groß ist :-))

Eine Ausnahme des Pagings sind XPK-gepackte Dateien, die der Einfachheit halber immer vollständig eingelesen werden. Für sie gibt es also **kein** Paging! (see Chapter 9 [XPK Schnittstelle], page 21)

Natürlich kann Most Texte mit allen ANSI-Steuersequenzen anzeigen. Dafür habe ich das Rad nicht noch einmal erfunden, sondern auf Bewährtes zurückge- griffen: Das console.device. Als Nebeneffekt kann man Textstellen markieren und in das Clipboard kopieren. Leider hat dies auch drei Nachteile:

- Man kann nur innerhalb der sichtbaren Seite markieren
- Das Scrollen ist auf Amigas ohne 680x0 ($x \geq 2$) relativ langsam ;-)
- Binäre Dateien sollte man NICHT einladen. Das gibt Chaos...

3 Konfiguration

Most konfiguriert sich in der Reihenfolge

1. Interne Vorgabewerte
2. Environmentvariablen `Most/Most` und `Most/FileReq` (Bei Aufruf in einer Shell werden lokale Variable zuerst eingelesen)
3.
 - Bei Shell-Start: Shellparameter auswerten
 - Bei WB-Start : Icon-Tooltypes auswerten
4. NEU seit Most 1.53: Zu jedem Text wird – sofern vorhanden – dessen Icon geöffnet und die Tooltypes ausgewertet. Die darin stehenden Werte gelten dann lokal für das neue Fenster. Außerdem können hier (und nicht im Most-Programmicon!) auch alle ARexx-Befehle direkt eingetragen werden. Diese werden dann ebenfalls der Reihe nach ausgeführt, quasi als eine Art "Tooltype-Skript" (see Chapter 11 [Der ARexx Port], page 23).

Folgende Optionen sind jeweils in 2.), 3.) und 4.) erlaubt:

3.1 FILES

Verwendung

Eine Liste von Dateien (nur beim Start aus einer Shell). Es können die bekannten AmigaDOS Wildcards verwendet werden. Seit Version 1.55 findet Most auch Dateien, deren Namen selbst AmigaDOS Wildcards, also zum Beispiel Klammern, enthalten.

3.2 PORTNAME

Verwendung

Name des globalen ARexx Ports von Most (see Chapter 11 [Der ARexx Port], page 23).

Vorgabewert

MOSTPORT

3.3 PUBSCREEN

Verwendung

Zur Angabe eines Bildschirms, auf dem Most sein Fenster öffnet.

3.4 STARTUP

Verwendung

Arexx Skript das beim Starten von Most ausgeführt wird (Achtung: Dieses Skript wird auch beim Öffnen jedes weiteren Textfensters ausgeführt!!) (see Chapter 11 [Der ARexx Port], page 23).

Alternativ können in die Tooltypes der Text-Icons auch direkt ARexx-Befehle geschrieben werden, die dann direkt nach dem Öffnen des Textes der Reihe nach ausgeführt werden.

3.5 WINDOWX

Abkürzung

WX

Verwendung

Zur Angabe der X-Koordinate der Most-Fenster.

3.6 WINDOWY

Abkürzung

WY

Verwendung

Zur Angabe der Y-Koordinate der Most-Fenster. -1 zentriert die Fenster.

3.7 WINDOWW

Abkürzung

WW

Verwendung

Die Breite der Fenster angeben. Wird ein negativer Wert oder Null angegeben, so wird die Breite des Bildschirms kopiert.

Vorgabewert

600

3.8 WINDOWH

Abkürzung

WH

Verwendung

Zur Angabe der Fensterhöhe. Wird ein negativer Wert oder Null angegeben, so wird die Höhe des Bildschirms (ohne Titel) kopiert.

Vorgabewert

400

3.9 SCREENMODE

Abkürzung

SM

Verwendung

Der Name einer Bildschirmauflösung, die dem Betriebssystem bekannt ist (Most verwendet keine fest einprogrammierte Namensliste). Eine Liste dieser Namen sieht man zum Beispiel im Fenster des **Prefs/ScreenMode** Programms. Wird ein gültiger Modus angegeben, so öffnet Most einen eigenen Bildschirm (Name: Most), auf dem seine Fenster liegen. **Achtung:** Je nach Version der Monitordateien ändern sich eventuell die Namen der Auflösungen.

Beispiel

```
SCREENMODE "Pal:HighRes"
```

3.10 FONT

Verwendung

Der Name der gewünschten Schriftart für den Text inklusive der Endung “.font”. Achtung mit Proportionalschriften! Die Größe der Schrift kann mit der Option Section 3.11 [FONTHEIGHT], page 8 angegeben werden, muß aber nicht.

Beispiel

```
FONT "helvetica.font"
```

3.11 FONTHEIGHT

Verwendung

Hiermit kann die Höhe einer gewählten Schriftart festgelegt werden. Wird hiermit die Höhe nicht explizit angegeben, so wird die Höhe des Standard-System-Fonts kopiert. Diese Option wirkt nur in Zusammenhang mit der Option Section 3.10 [FONT], page 7.

3.12 APPICON

Verwendung

Chapter 8 [Das AppIcon], page 20 von Most wird mit dieser Option angeschaltet. Wird Most mit dieser Option ohne weitere Dateinamen gestartet, dann wartet das Programm im Hintergrund und öffnet nur das AppIcon.

3.13 ICONX

Verwendung

Die X-Koordinate für das Section 3.12 [APPICON], page 8 auf der Workbench.

3.14 ICONY

Verwendung

Die X-Koordinate für das Section 3.12 [APPICON], page 8 auf der Workbench.

3.15 BUFLEN

Verwendung

Hiermit läßt sich die maximale Länge des Lesespeichers angeben. Für kleiner Dateien wird nur ein Puffer von der Länge der Datei allokiert. Dadurch kann mit sehr großen Puffergrößen erzwungen werden, alle Dateien vollständig einzuladen.

Vorgabewert

50 000 Bytes

3.16 PIPELEN

Verwendung

Most kann auch aus einer Pipe lesen. Normalerweise werden dann fest Section 3.15 [BUFLEN], page 9/10 Bytes zum Lesen belegt. Mit dieser Option kann man die Speichergröße für Pipes explizit setzen.

Beispiel

PIPELEN 8192

3.17 QUOTES

Verwendung

Hiermit wird eine Folge von Trennzeichen definiert. Der Sinn dahinter ist die Möglichkeit, mit Doppelklick auf ein Wort, das **zwischen** den hiermit angegebenen Trennern steht, einen Text mit dem entsprechenden Namen zu laden. Das ist besonders für Programmierer gedacht, die somit leicht die Inklude-Dateien in ihren Sourcen anspringen können.

Vorgabe

"'<>(){}[]

3.18 DIRS

Verwendung

Hiermit können mehrere Pfade angegeben werden, die der Reihe nach durchsucht werden, wenn Most nach einem Doppelklick in den Text Inklude-Dateien einzuladen ver-

sucht (siehe Section 3.17 [QUOTES], page 9). Die einzelnen Pfade werden durch das Zeichen | voneinander getrennt.

Beispiel

```
DIRS "Include:|TexInputs:|LC:Include/"
```

3.19 ENDS

Verwendung

Endungen, die automatisch an Inklude-Dateien (der Reihe nach) angehängt werden (vgl. Section 3.18 [DIRS], page 9, Section 3.17 [QUOTES], page 9).

Beispiel

```
ENDS ".c.h.tex"
```

3.20 STACKSIZE

Verwendung

Dient dazu, um dem Editor, der mit dem aktuellen Text gestartet werden kann, eine bestimmte Stackgröße einzustellen. Dies ist zum Beispiel für den Emacs ("Eight Megabytes And Constantely Swapping"?) nützlich.

Beispiel

```
STACKSIZE 50000
```

Vorgabe 8192 Bytes

4 Tastenbefehle

Grundsätzlich werden alle von More her bekannten Tastenkommandos unterstützt. Die Befehle unterteilen sich in folgende Bereiche:

4.1 Scrollen

eine Zeile tiefer

CURSOR RUNTER, ZIFFERBLOCK 2, RETURN, ENTER

eine Zeile hoch

CURSOR HOCH, ZIFFERBLOCK 8, DELETE

eine Seite weiter springen

SHIFT CURSOR RUNTER, SPACE, PAGE DOWN

Eine Seite zurück springen

SHIFT CURSOR HOCH, BACKSPACE, PAGE UP, b

Zum Textanfang springen

HOME, <

Zum Textende springen

END, >

Eine Spalte nach rechts

CURSOR RECHTS, ZIFFERBLOCK 6

Eine Spalte nach links

CURSOR LINKS, ZIFFERBLOCK 4

8 Spalten nach rechts

SHIFT CURSOR RECHTS

8 Spalten nach links

SHIFT CURSOR LINKS

Zum Spaltenanfang springen

ZIFFERBLOCK 5

4.2 Suchbefehle

Bisherigen Suchbegriff weiter suchen (vorwärts)

n

Suchen ohne auf Groß/Kleinschrift zu achten

.

Suchen mit Unterscheidung nach Groß/Kleinschrift

/

DAS SUCHFENSTER öffnen

f, s

4.3 Laden

Neuen Text laden mit **l** (Dateiauswahlfenster erscheint).

4.4 Informationsfenster

Neben dem üblichen Chapter 1 [Copyright], page 2 stehen hier nähere Informationen zum aktuellen Text: Der Dateiname, die Dateigröße, die Zeilenzahl und die davon gerade sichtbaren Zeilen, sowie der Name des XPK Packers (oder einfach nur “—” wenn die Datei ungepackt ist). Hier erscheint auch entweder die Meldung, daß der Text komplett eingelesen wurde, oder die Größe des benutzten Lesespeichers.

Das Informationsfenster öffnen Sie mit **h**.

4.5 Hilfe

Wenn Sie **HELP** drücken, wird der AmigaGuide mit der Beschreibung von Most gestartet. Dazu muß natürlich der AmigaGuide installiert sein. Wenn Most die `amigaguide.library` nicht öffnen kann, dann erscheint stattdessen das Section 4.4 [Informationsfenster], page 14.

Der AmigaGuide sucht seine Dokumente im aktuellen Verzeichnis und in allen Pfaden, die in der Environment Variablen *AmigaGuide/Path* aufgelistet sind.

4.6 Editieren

Wenn Sie **e** drücken, wird der Text in einen Editor geladen. Der Editor wird asynchron gestartet. Den Namen des Editors erwartet Most in der Environmentvariablen **EDITOR**.

Hinweis: Die Stackgröße für den Editors kann über die Option Section 3.20 [STACKSIZE], page 12 eingestellt werden.

4.7 Beenden

Das Textfenster wird mit `ESCAPE`, `CTRL-C` oder `q` geschlossen.

5 Menus

Es gibt folgende Hauptmenüs in Most:

5.1 Projekt

- Neu...** Öffnet ein neues Fenster und fragt per Dateiauswahlfenster nach einer neuen Datei. Wenn der Dateiauswahlfenster abgebrochen wird, so wird das Fenster wieder geschlossen.
- öffnen...** Per Dateiauswahlfenster den Namen einer neuen Textdatei auswählen und in das aktuelle Fenster laden. Die Ausmaße, Koordinaten und das Pattern des Dateiauswahlfensters können im Section 5.5 [Einstellungen], page 17 Menü abgespeichert werden.
- editieren...** Startet den Editor mit dem aktuellen Text (per RUN). Der Name des Editors muß in der Environment Variable EDITOR stehen, die Größe seines Stacks kann mit der Option Section 3.20 [STACKSIZE], page 12 eingestellt werden.
- drucken** Druckt den aktuellen Text über den Systemdruckertreiber (PRT:) aus.
- Info...** Öffnet ein Section 4.4 [Informationsfenster], page 14, in dem neben dem üblichen Chapter 1 [Copyright], page 2 einige wichtige Informationen zum aktuellen Text stehen: Name, Dateilänge, Zeilenzahl, sichtbare Zeilennummern, XPK-Packername und entweder die Meldung, daß der Text komplett eingelesen wurde, oder die Größe des benutzten Puffers.
- Ende** Beendet den aktuellen Text und — falls es der letzte Text ist — auch ganz Most.

5.2 Bewegen

Verschiedene Menüpunkte zum zeilen- und seitenweisen Springen im Text, bzw. um gezielt die letzte oder erste Seite anzuspringen.

Mit dem Menü “zu Zeile ...” kann man in einem Dialogfenster eine bestimmte Zeilennummer eingeben, zu der man im Text Springen möchte. Most ignoriert ungültige Zeilennummern (kleiner 1 oder zu groß).

5.3 Suchen

suchen... Chapter 7 [Das Suchfenster], page 19 wird geöffnet und aktiviert dessen Texteingabefeld. Bei bereits offenem Suchfenster ist dieser Menüpunkt abgeschaltet.

vorwärts/rückwärts

Suchen des eingegebenen Suchtextes in die jeweilige Richtung. Beide Punkte sind nur anwählbar, wenn das Suchfenster geöffnet ist.

5.4 Extras

Bild neu aufbauen

Initialisiert die Console neu und baut die aktuelle Seite noch einmal vollständig auf.

Tabulatorbreite einstellen...

Öffnet ein Dialogfenster, in dem die Breite der Tabulatoren eingegeben werden kann. Normalerweise sind die Tabulatoren alle 8 Zeichen gesetzt.

Skript starten...

Öffnet einen Dateiauswahlfenster, mit dem ein ARexx Makro ausgewählt werden kann, das anschließend ausgeführt wird.

5.5 Einstellungen

sichern

global Speichern der momentanen Einstellungen in Env: und EnvArc: in ein Unterverzeichnis **Most** (das automatisch erzeugt wird). Es werden alle von der Kommandozeile bekannten Optionen gespeichert. Zusätzlich werden in einer extra Variablen die Maße, die Position und das Pattern des Dateiauswahlfensters abgespeichert.

lokal Speichern der Einstellungen in zwei lokale Shell Variablen namens *Most*/*Most* und *Most/FileReq* (sorry wegen den Namen...). Diese Option funktioniert natürlich nur, wenn Most von einer Shell aus gestartet worden ist!

6 Das Textfenster

Titelzeile Links steht der Dateiname, möglichst weit rechts wird die aktuelle Spalten- und Zeilennummer der obersten (sichtbaren) Zeile angegeben. Format:

<Dateiname>

<Spalte>/<Zeile>

Symbole Neben den bekannten Systemsymbolen zum Schließen, Verkleinern usw. gibt es drei weitere Symbole am rechten Fensterrand, mit denen im Text gescrollt werden kann. Der Scrollbalken kann sowohl frei bewegt, als auch durch hineinklicken seitenweise benutzt werden. Die Balkenposition gibt die Position im Text, seine Größe das Verhältnis der sichtbaren Seite zum ganzen Text an. Die beiden Pfeilsymbole darunter dienen zum zeilenweisen Auf- und Abscrollen.

Maussteuerung

Wir mit der Maus in das Fenster doppelt geklickt, so wird an der entsprechenden Stelle nach einem Dateinamen zwischen den konfigurierten Trennzeichen (see Section 3.17 [QUOTES], page 9) gesucht. Ist dort eine solche Textstelle vorhanden (z.B. in C-Quelltexten ein `#include <exec/exec.h>` mit dem Dateinamen `exec/exec.h` zwischen `<` und `>`), dann wird jener in allen konfigurierten Verzeichnissen (vgl. Section 3.18 [DIRS], page 9) gesucht und sofern gefunden in einem eigenen Fenster angezeigt.

Clipboard Es kann aus den Fenstern durch das einfache Anklicken des Textes und dem anschließenden Bewegen der Maus (**ohne** den linken Knopf loszulassen!) ein Textbereich im Fenster markiert werden. Der markierte Textbereich kann dann mit **AMIGA-C** kopiert werden.

AppWindow

Jedes Most-Textfenster ist ein sogenanntes AppWindow, in das jederzeit eines oder auch mehrere Icons gezogen werden können. Die erste Datei wird dann in das aktuelle Fenster eingeladen und für alle weiteren Dateien werden eigene Fenster geöffnet.

7 Das Suchfenster

Im Suchfenster kann ein Suchbegriff eingegeben, sowie die drei Suchoptionen eingestellt werden. Dies geht sowohl mit der Maus als auch per Tastatur. Die entsprechenden Buchstabenabkürzungen sind in den Symboltexten unterstrichen. Außerdem hat man im Suchfenster die gleiche Funktionalität wie im zugehörigen Textfenster:

- Es sind dieselben Menüs und deren Shortcuts vorhanden. Lediglich der Menüpunkt Section 5.3 [Suchen], page 17 ist abgeschaltet und dafür die Punkte vorwärts/rückwärts suchen zugelassen.
- Neben den speziellen Tastenbefehlen für die Symbole des Suchfensters werden **alle TASTENBEFEHLE** des Textfensters weiterhin unterstützt. Insbesondere kann man also bei aktiviertem Suchfenster mit den Cursortasten im Textfenster scrollen!!! (Achtung: Das geht nur bei inaktivem Texteingabefeld)

8 Das AppIcon

Das AppIcon erlaubt zwei Funktionen:

Doppelklick

Ein Dateiauswahlfenster erscheint, in dem eine Datei ausgewählt werden kann. Ist nur noch das AppIcon von Most aktiv (das heißt keine Textfenster offen) und wird nun das Auswahlfenster abgebrochen, so wird damit Most beendet (vorher kommt natürlich noch eine Sicherheitsabfrage)!

Icons

Werden Piktogramme auf das AppIcon gezogen, so werden jene angezeigt.

Bemerkungen:

Most verwendet immer das eigene Piktogramm als Section 3.12 [APPICON], page 8! Ein eingebautes Bild bzw. ein explizit angegebenes werden nicht angeboten/unterstützt. Die Koordinaten des AppIcons werden im Menü Section 5.5 [Einstellungen], page 17 mit abgespeichert. Mit den Optionen Section 3.13 [ICONX], page 8 und Section 3.14 [ICONY], page 8 können explizite Koordinaten angegeben werden.

9 XPK Schnittstelle

Die Schnittstelle zum XPK Standard ist absolut transparent, um nicht zu sagen, unsichtbar gehalten. Ist eine Datei mit einem XPK Packer gepackt und ist das XPK Packet korrekt installiert, so entpackt Most die Datei automatisch und zeigt sie dann an. Im Section 4.4 [Informationsfenster], page 14 erscheint nun hinter “XPK-Packer:” nicht mehr wie üblich “—”, sondern der Name des verwendeten Packers. Einzige Besonderheit: Die Datei wird vollständig eingeladen, ein “Paging” findet nicht statt!

Trifft Most auf eine mit XPK verschlüsselte Datei, dann öffnet er erst ein Passwort-Dialogfenster, in dem man sein Passwort angeben kann. Aber Vorsicht: Man hat nur diese eine Gelegenheit! Gibt man ein falsches Passwort an, sieht man den verschlüsselten Kauderwelch und muß den Text erneut laden.

Most kann ohne Probleme auch ohne XPK verwendet werden. Die xpkmaster.library ist also nicht nötig, um mit Most Texte anzuzeigen. Ohne sie werden XPK gepackte Texte dann natürlich auch nicht entpackt...

10 Pipe Verwendung

Neuerdings unterstützt Most auch Pipes. Diese Option ist aber immer noch wenig getestet. Bitte meldet mir alle Fehler.

Wurde Most zum Beispiel mit

```
type xyz | Most
```

gestartet, dann öffnet es ein Textfenster Namens **<Pipe Inhalt>** und zeigt den Inhalt der Pipe an. Ist jener kleiner als die konfigurierte Puffergröße, dann kann ganz normal darin gescrollt werden (editieren funktioniert aber nicht!). Ist der Puffer zu klein und scrollt man über den Puffer hinaus, so wird zwar weitergelesen, ein zurück zum Dateianfang gibt es jetzt aber nicht mehr. Wie üblich kann man Most natürlich einen größeren Puffer mitgeben...

Kleine Einschränkungen:

Der Puffer wird mit einer Größe von Section 3.15 [BUFLEN], page 9 / 10 angelegt (wie soll man einer Pipe entlocken, wieviel sie enthält/enthalten wird?). Mit der Option Section 3.16 [PIPELEN], page 9 kann man explizit die Speichergröße für Pipes festlegen.

XPK aus einer Pipe ist nicht möglich.

11 Der ARexx Port

Neben dem globalen ARexx-Port *MOSTPORT* (bzw. der durch die Option Section 3.2 [PORT-NAME], page 5 angegebene Name) des Hauptprozesses von Most besitzt jedes Textfenster seinen eigenen ARexx Port. Um ein spezielles Textfenster anzusprechen, sendet man an den globalen ARexx Port zuerst den Befehl Section 11.1 [GetFileList], page 23, der eine Liste aller geöffneten Texte und deren Portnamen zurückgibt. Anhand jener Liste kann dann das gewünschte Fenster direkt adressiert werden. Während man an die Ports der einzelnen Fenster **alle** ARexx-Befehle senden kann, “versteht” der Hauptport nur die ersten 6 Befehle Section 11.1 [GetFileList], page 23, Section 11.2 [GetScreenName], page 24, Section 11.3 [Quit], page 24, Section 11.6 [OpenNew], page 25, Section 11.5 [Set], page 25 und Section 11.4 [Request], page 24.

Most setzt in RC eine Fehlernummer von

- 0, wenn der Befehl korrekt ausgeführt wurde.
- 5, wenn Section 11.7 [Open], page 25 oder Section 11.6 [OpenNew], page 25 fehlschlagen
- 10, bei Fehlern in den Parametern zu einem Befehl.
- 20, falls ein unbekanntes Befehlswort gesendet wurde.

Folgende ARexx Befehle werden unterstützt:

11.1 GetFileList

Aufruf `GetFileList`

Resultat In *RESULT* wird eine Liste aller vorhandenen Textfenster und ihrer Portnamen zurückgegeben. *RESULT* hat den Aufbau: {<Fenstername> <Portname> }⁺ (das heißt mindestens einmal aber beliebig viele)

Beispiel

```
GetFileList
RESULT: "Most.dok MOST.0 Most.doc MOST.1"
```

11.2 GetScreenName

Aufruf `GetScreenName`

Resultat **RESULT** enthält den Schirmnamen

Wirkung Liefert den Namen des Bildschirms, auf dem Most den aktuellen Text darstellt.

Beispiel

```
GetScreenName
RESULT - "Workbench"
```

11.3 Quit

Aufruf **Quit**

Wirkung Beendet Most und schließt **alle** Fenster. Chapter 8 [Das AppIcon], page 20 von Most wird auch entfernt – sofern vorhanden.

Beispiel

```
Quit
```

11.4 Request

Aufruf **Request** TITLE/K/A TEXT/K/A GADGETS/K/A

Resultat **RESULT** enthält Nummer des gedrückten Gadgets:

0 das am weitesten rechts (meist Abbrechen)

1 bis n-1 die restlichen von links durchnummeriert

Wirkung Eine einfache Möglichkeit einer Benutzerabfrage per Dialogfenster. Titel und Text sind einfache Strings. Mit **GADGETS** werden die Texte der Gadgets angegeben, durch '|' getrennt.

Beispiel

```
Request TITLE "Test" TEXT "Klaro?" GADGETS "JA|Nein"
RESULT - 1 wenn JA gewählt wurde, sonst 0
```

11.5 Set

Aufruf `Set CASE/S WORD/S BACK/S`

Wirkung Durch das Angeben eines der Schlüsselworte wird die entsprechende Option angeschaltet, während die Optionen, die nicht angegeben wurden, implizit ausgeschaltet werden. Die Parameter bedeuten

`CASE` casesensitives Suchen an

`WORD` wortweises Suchen an

`BACK` Rückwärtssuchen an

Beispiel

`Set WORD ON`

Das wortweise Suchen wird an-, die anderen beiden Suchoptionen ausgeschaltet.

11.6 OpenNew

Aufruf `OpenNew FILE/K`

Wirkung In ein eigene Fenster einen neuen Text laden. Die Angabe eines Dateinamens ist optional. Wird er weggelassen, so wird ein Dateiauswahlfenster geöffnet.

Es braucht vorher kein aktuelles Fenster gewählt worden sein. Das Kommando kehrt erst zurück, wenn der Text beendet wurde!

Beispiel

`OpenNew FILE Most.dok`

11.7 Open

Aufruf `Open FILE/K`

Wirkung In das aktuelle Fenster eine neue Datei einladen. Die Angabe eines Dateinamens ist optional. Wird er weggelassen, so wird ein Dateiauswahlfenster geöffnet. Das Kommando kehrt erst zurück, wenn der Text beendet wurde!

Beispiel

`Open FILE Most.dok`

11.8 Close

Aufruf Close

Wirkung Schließt das aktuelle Fenster.

11.9 ToFront

Aufruf ToFront

Wirkung Holt das aktuelle Fenster nach vorne, so daß es vollständig sichtbar wird. Dazu wird auch der Bildschirm, auf dem sich das Fenster befindet, nach vorne geholt und aktiviert.

11.10 GetWindowDims

Aufruf GetWindowDims

Resultat In RESULT steht <xpos> <ypos> <breite> <höhe>

Wirkung Liefert die Ausmaße des akt. Fensters in RESULT

Beispiel

```
GetWindowDims
RESULT - "40 40 600 180"
```

11.11 Edit

Aufruf Edit

Wirkung Section 4.6 [Editieren], page 14 des Textes des aktuellen Fensters (Editor starten).

11.12 Info

Aufruf Info

Wirkung Öffnet das Section 4.4 [Informationsfenster], page 14 zum aktuellen Text.

11.13 FirstPage

Aufruf FirstPage

Wirkung Die erste Seite des Textes anzeigen

11.14 LastPage

Aufruf LastPage

Wirkung Die letzte Seite des Textes anzeigen.

11.15 NextPage

Aufruf NextPage

Wirkung Nächste Seite des aktuellen Textes anzeigen.

11.16 PrevPage

Aufruf PrevPage

Wirkung Springt zur vorherigen Seite im Text.

11.17 NextLine

Aufruf NextLine

Wirkung Eine Zeile weiterscrollen

11.18 PrevLine

Aufruf PrevLine

Wirkung Eine Zeile zurückscrollen

11.19 NextColumn

Aufruf NextColumn

Wirkung Eine Spalte nach rechts scrollen

11.20 PrevColumn

Aufruf PrevColumn

Wirkung Eine Spalte nach links scrollen (bis maximal zum Textanfang)

11.21 NextTab

Aufruf NextTab

Wirkung Um 8 Spalten nach rechts scrollen

11.22 PrevTab

Aufruf PrevTab

Wirkung Um 8 Spalten nach links scrollen (bis maximal zum Textanfang)

11.23 Goto

Aufruf Goto LINE/K/N/A COLUMN/K/N

Wirkung Als Argument **muß** eine Zeile und optional kann die Startspalte angegeben werden. Die Zeilennummer wird als neuer Seitenbeginn genommen.

Beispiel

Goto LINE 142

11.24 Search

Aufruf Search TEXT/K

Wirkung Ohne Argument: Chapter 7 [Das Suchfenster], page 19 öffnen, bzw. bereits eingegebenen Text weitersuchen

 Mit Suchbegriff: Als neuen Suchbegriff einsetzen und suchen.

Beispiel

 Search TEXT huhu

12 Bekannte Fehler

- Verschiedene Kontrollsequenzen zum Setzen des Cursors sorgen für Chaos in der Anzeige.
- Die Chapter 10 [Pipe Verwendung], page 22 hat Probleme mit der WShell. Jene scheint immer nur zeilenweise in die Pipe zu schreiben, was Most fehlinterpretiert, da es weniger Daten liest als es wollte (nämlich seinen ganzen Pipe-Puffer voll). Sollte ich bei Gelegenheit mal ändern. . .
- Zumindest zu Most 1.53 wurden mir Enforcerhits gemeldet, die wohl auf ein Überlaufen meines Eingabepuffers zurückgehen. Mit der jetzigen Version konnte ich diesen Fehler nicht mehr reproduzieren, kann ihn aber auch nicht ausschliessen.
- Bruno Lacombe hat mir einen Mungwall-Hit von Most 1.53 gemeldet, den ich aber auch nicht nachvollziehen kann. Vielleicht kann mir ein anderer genaueres melden?!?

13 Kritiken - Ideen

Ich erweitere Most eigentlich fortwährend und versuche Fehler so schnell wie möglich zu beseitigen. Es gibt auch noch ein paar Ecken und Kanten, deren ich mir bewußt bin. Ich kann jedoch nicht versprechen, daß ich sie in naher Zukunft beheben werde. Schließlich arbeite ich an Most in meiner Freizeit und verdiene kein Geld damit. . .

Nichtsdestoweniger bin ich sehr an Anregungen und auch jeglicher Kritik und Bugreports interessiert! Schreibt mir, ruft mich an oder kommt vorbei, nur teilt es mit mit :-) Most lebt von Eurer Mitwirkung und die meisten Anregungen haben ich auch übernommen!

Meine Adresse:

Uwe Röhm
Wörthstr. 18
94 032 Passau

EMail:

bugs@roemer.deg.sub.org

Fido:

2:2494/22.5

14 aktuelle Version

Die aktuellste Most Version ist immer per ftp auf `ftp.uni-passau.de` (132.231.1.10) in `pub/amiga/local/UweR` zu beziehen. Tip: In dem Verzeichnis `pub/amiga/local` liegen noch ganz andere interessante Programme :-)

Ich werde versuchen, die aktuellen Releases immer auf das AmiNet und auf “Die Box Passau” (Fido-Mailbox, 2:2494/22.0) hochzuspielen.

15 Danksagungen

Inzwischen schreibe ich schon rund zwei Jahre an Most und immer noch finden sich alte (und neue) Fehler. Die meisten wären mir selbst entgangen, wenn ich nicht so viele Mails von Euch, den Benutzern, bekommen würde! Ich kann hier und jetzt nicht alle Namen auflisten, will aber zumindest die Emsigsten unter Euch nennen:

Karl "Charly" Skibinski, David Göhler, Uli Zukowski, Stefan Sticht,
Matthias "Tron" Scheler, Jan Holler, Markus Uhlendahl und Dirk Federlein

Wer mit Übersetzungen schickt, sollte dies entweder per Diskette tun, oder bei Mails ein gängiges **uuencode** verwenden! Mir wurden jetzt schon oft Übersetzungen als Mail geschickt, die fehlerhaft ankamen. Das muß nicht sein und ist schade um Eure Mühe! Die bisherigen Übersetzungen stammen von:

- Volker-Alexander Graf (dänisch)
- Frank Geider (französisch)
- Francesco Dipietromaria (italienisch)
- Magnus Holmgren (schwedisch)
- Antonio Joaquin Gomez Gonzalez (spanisch)

Zum Schluß ein Dankeschön an Urban für seine Mühe und XPK. Und außerdem ein herzliches Dankeschön an alle, die mir Bugreports, Vorschläge und natürlich Geschenke geschickt haben!

Index

A

AINSI Sequenzen	3
aktuelle Version	32
AmigaGuide	14
Anregungen	31
AppIcon	20
AppWindow	18
ARexx	23
Aufruf von Most	5
Autor	31

B

Bewegen	16
---------------	----

C

Copyright	2
-----------------	---

D

Danksagungen	33
--------------------	----

E

Editieren	14
Einstellungen	17
Environment Variablen	5
Extras	17

F

Fehler	30
Fenster	18
Fensterverwaltung	3

G

gepackte Dateien	21
Giftware	2

H

Hilfe	14
-------------	----

I

Ideen	31
Includedateien anspringen	18
Informationsfenster	14

K

Konfiguration	5
Kritik	31

M

Makros	17
Markieren	18
Maussteuerung	18
Menus	16
Merkmale	5

P

Paging	3
Parameter	5
Pipe Verwendung	22
Prinzip	3
Projekt	16
Prozeße	3

S

Shell Aufruf	5
Skripte	23
Speicherverwaltung	3
Suchbefehle	13
Suchen	17, 19
Suchfenster	19
Symbole	18

T

Tastenbefehle	13
Textfenster	18
Titelzeile	18
ToolTypes	5

V

Variablen	5
verschlüsselte Dateien	21
Vorgabewerte	5

W

Workbench Start	5
-----------------------	---

X

XPK	21
-----------	----

Table of Contents

1	Copyright	2
2	Prinzip	3
3	Konfiguration	5
3.1	FILES	5
3.2	PORTNAME	5
3.3	PUBSCREEN	6
3.4	STARTUP	6
3.5	WINDOWX	6
3.6	WINDOWY	6
3.7	WINDOWW	6
3.8	WINDOWH	7
3.9	SCREENMODE	7
3.10	FONT	7
3.11	FONTHEIGHT	8
3.12	APPICON	8
3.13	ICONX	8
3.14	ICONY	8
3.15	BUFLEN	9
3.16	PIPELEN	9
3.17	QUOTES	9
3.18	DIRS	9
3.19	ENDS	11
3.20	STACKSIZE	12
4	Tastebefehle	13
4.1	Scrollen	13
4.2	Suchbefehle	13
4.3	Laden	14
4.4	Informationsfenster	14
4.5	Hilfe	14
4.6	Editieren	14
4.7	Beenden	15
5	Menus	16
5.1	Projekt	16

5.2	Bewegen	16
5.3	Suchen	17
5.4	Extras	17
5.5	Einstellungen	17
6	Das Textfenster	18
7	Das Suchfenster	19
8	Das AppIcon	20
9	XPK Schnittstelle	21
10	Pipe Verwendung	22
11	Der ARexx Port	23
11.1	GetFileList	23
11.2	GetScreenName	23
11.3	Quit	24
11.4	Request	24
11.5	Set	25
11.6	OpenNew	25
11.7	Open	25
11.8	Close	26
11.9	ToFront	26
11.10	GetWindowDims	26
11.11	Edit	26
11.12	Info	26
11.13	FirstPage	27
11.14	LastPage	27
11.15	NextPage	27
11.16	PrevPage	27
11.17	NextLine	27
11.18	PrevLine	27
11.19	NextColumn	28
11.20	PrevColumn	28
11.21	NextTab	28
11.22	PrevTab	28
11.23	Goto	28
11.24	Search	29

12	Bekannte Fehler.....	30
13	Kritiken - Ideen.....	31
14	aktuelle Version.....	32
15	Danksagungen.....	33
	Index.....	34