

AssignList

Markus Adamski

COLLABORATORS

	<i>TITLE :</i> AssignList		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Markus Adamski	July 20, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	AssignList	1
1.1	AssignList - Übersicht	1
1.2	Was ist AssignList ?	1
1.3	Systemanforderungen	1
1.4	Wie wird AssignList installiert ?	2
1.5	AssignList starten	2
1.6	Hauptfenster	3
1.7	Liste aller definierten Assigns	4
1.8	Assign-Name	4
1.9	Assign-Pfad	4
1.10	Assign-Modus	4
1.11	Assign ausführen	5
1.12	Teilliste ausführen	5
1.13	Liste aller temporären Assigns	6
1.14	Alle temporären Assigns ausführen	6
1.15	Markierten temporären Assign ausführen	6
1.16	Markierten temporären Assign löschen	6
1.17	Liste der temporären Assigns an Hauptliste anhängen	6
1.18	Liste temporärer Assigns löschen	6
1.19	Aktuellen Definition zur Liste hinzufügen	7
1.20	Markierten Definition löschen	7
1.21	Liste speichern	7
1.22	Liste speichern und Programm beenden	7
1.23	Programm beenden	7
1.24	Aktuelle Datei	7
1.25	Das Menü	7
1.26	Format der Definitionsdatei	8
1.27	Was noch zu tun ist...	8
1.28	ARexx-Unterstützung	9
1.29	ARexx: AddItem	9
1.30	Über MUI	9

Chapter 1

AssignList

1.1 AssignList - Übersicht

```
AssignList

Assigns GUI-gesteuert definieren

Copyright © 1996 von Markus Adamski

- Freeware -

Was ist AssignList ?
Systemanforderungen
Wie wird AssignList installiert ?

AssignList starten
Hauptfenster
Menü

ARexx-Unterstützung

Format der Definitionsdatei

Programm-Geschichte
Was noch zu tun ist...

Über MUI
```

1.2 Was ist AssignList ?

Mit AssignList ist es möglich, beliebig viele Assigns während des Boot-Vorganges durch einen einzigen Aufruf auszuführen.

1.3 Systemanforderungen

AssignList benötigt mindestens OS 2.04 oder höher und MUI in der Version 3.1 oder höher, um in vollem Umfang genutzt werden zu können.

Um Assigns während des Boot-Vorganges auszuführen, ist lediglich die dos.library ab Version 37 erforderlich.

Die Anforderungen an die CPU sind minimal, ein 68000 genügt.

1.4 Wie wird AssignList installiert ?

Doppelklick im Verzeichnis Install auf das Installer-Icon in der gewünschten Sprache.

Damit werden alle Dateien kopiert und die Startup-Sequence modifiziert.

1.5 AssignList starten

Es gibt zwei Möglichkeiten, AssignList zu starten:

1. Aufruf von der Shell bzw. in der Datei User-Startup:

- Einfach "AssignList" in der Shell eingeben bzw. in der User-Startup angeben und AssignList startet.
Hierbei werden noch einige Parameter unterstützt:

- * HELP bzw. ?
Ausgabe eines kurzen Hilftextes.
- * <Dateiname>
Nach Programmstart wird die Datei <Dateiname> automatisch geladen.
- * <Dateiname> USE
Die Datei <Dateiname> wird geladen, die darin enthaltenen Assign-Definitionen ausgeführt und das Programm wieder beendet.
Hierfür ist MUI nicht erforderlich !
Dies ist außerdem der Standardeintrag in der User-Startup, der bei der Installation erzeugt werden kann.

2. Aufruf von der Workbench

- Einfach auf das Icon von AssignList doppelklicken und das Programm wird gestartet. Es kann auch AssignList ins Verzeichnis WBStartup plaziert werden, um bei jedem Neustart geladen zu werden.
Hier werden folgende Parameter als ToolType unterstützt:
-

- * DONOTWAIT
Falls AssignList im Verzeichnis WBStartup liegt, wird dem System mitgeteilt, nicht auf die Beendigung von AssignList zu warten.
(Kann ruhig immer angegeben werden.)
- * CX_POPKEY=<Tastenkombination>
Legt eine Tastenkombination fest, mit welcher das Hauptfenster geöffnet werden kann, falls AssignList iconifiziert ist.
- * CX_POPUP=<YES|NO>
Legt fest, ob nach dem Programmstart das Hauptfenster geöffnet werden soll oder nur ein Icon auf der Workbench erscheint.
- * FILE=
Die Datei nach "=" wird beim Programmstart gelesen.
- * MAKEICONS=<YES|NO>
Legt fest, ob beim Speichern einer Definitionsdatei für diese Datei ein Icon erstellt werden soll oder nicht. Ein Doppelklick auf dieses Icon würde dann AssignList starten und diese Datei würde geladen.
(Die entspricht dem Menüpunkt "Icon erzeugen".)

1.6 Hauptfenster

Das Hauptfenster hat folgenden Aufbau:

```
|
|                                     |
|               Liste aller definierten Assigns           |
|               |
|               |
| -----|
|          Assign:                               Modus    |
|              nach:                              |
Assign-Arten |
| -----|
|      Ausführen...       Hinzufügen             Löschen   |
| -----|
|          Speichern
Speichern & Ende     Abbruch  |
| -----|
|                                Aktuelle Datei            |
```

Bleibt der Mauszeiger einige Sekunden über einem der Felder unbewegt stehen, so erscheint ein kurzer Hilfstext.

1.7 Liste aller definierten Assigns

Hier befinden sich alle hinzugefügten Definitionen.

In der ersten Spalte steht der Modus des Assign. Einzelne Einträge können mit der Maus an beliebige Positionen innerhalb der Liste verschoben werden. Ein einfacher Klick auf einen Eintrag markiert diesen. Ein Doppelklick auf einen Eintrag bewirkt, daß Name, Pfad und Modus unterhalb der Liste gesetzt werden.

In der zweiten Spalte steht der Name, welcher dem Pfad in der dritten Spalte zugeordnet werden soll.

1.8 Assign-Name

Hier wird die Bezeichnung eingegeben, unter dem ein Pfad angesprochen werden soll.

Dieser darf maximal 31 Zeichen lang sein.

1.9 Assign-Pfad

Hier wird der Pfad eingegeben, welcher durch einen Namen angesprochen werden soll.

Dieser darf maximal 127 Zeichen lang sein.

1.10 Assign-Modus

Hier wird der Modus des Assigns festgelegt:

Locked = Die üblichste Methode. Ein Pfad wird einfach einem Namen zugeordnet. Eine evtl. vorhandene alte Definition wird überschrieben.

Add = Hier wird ein Pfad zu einem bereits definierten Namen hinzugefügt. Somit können unter einem einzigen Namen mehrere Pfade angesprochen werden, die scheinbar einem einzigen Pfad entsprechen.

Z.B.: Assign Locked Test: RAM:

Assign Add Test: Work:

Ein "Dir Test:" würde nun den Inhalt von RAM: und Work: ausgeben, als ob es ein Verzeichnis wäre.

Beim Speichern werden die Daten in das mit Assign Locked festgelegte Verzeichnis geschrieben.

Path = Dies ist eine unverbindliche Definition, jedoch ähnlich dem Modus Locked. Ein Name wird einem Pfad

zugeordnet, jedoch bei jeder Benutzung des Namens wird der zugehörige Pfad neu identifiziert.

Z.B.: Assign Path FONTS: DF0:FONTS

Ein "Dir FONTS:" zeigt den Inhalt des FONTS-Verzeichnisses der gerade eingelegten Diskette an und verlangt nicht die, welche beim Aufruf des Assigns im Laufwerk war.

Late = Hiermit wird eine Name einem Pfad ähnlich dem Modus Locked zugeordnet, jedoch wird erst dann verifiziert, ob der Pfad existiert, wenn das erstmal mittels z.B. "Dir Name:" auf die Definition zugegriffen wird.

1.11 Assign ausführen

Ein Requester öffnet sich, der folgende Optionen bietet:

1. Eingabe:

Es wird der Assign ausgeführt, der durch den Pfad , Namen und Modus bezeichnet wird.

2. Alles:

Die gesamte Liste wird ausgeführt.

3. Teil

4. Abbruch

Schließt den Requester ohne weitere Aktion.

1.12 Teilliste ausführen

Das Ausführen-Fenster hat folgenden Aufbau:

```

.-----.
|           | Alle           |
|           | Markierten    |
| Liste aller temporären Assigns | Löschen |
|           |-----|
|           | Anhängen      |
|           | Ende          |
.-----.

```

Es können Einträge aus der Liste im Hauptfenster mit der Maus in diese Liste gezogen werden. Dabei ist wie folgt vorzugehen:
 Ein Eintrag muß angeklickt werden und die linke Maustaste gedrückt gehalten werden. Nun muß der Mauszeiger links oder rechts über den Rand des ListView-Gadgets weggezogen werden. Der Eintrag erscheint nun nicht mehr verwischt. Jetzt kann durch Loslassen der linken Maustaste, wenn der Mauszeiger über dem obigen ListView-Gadget ist, dieser Eintrag ins Gadget übernommen werden.

Um Einträge von dieser Liste in die Hauptliste zu ziehen, muß umgekehrt verfahren werden.

Aufgepaßt: Es wird nicht geprüft, ob ein Name, der so einer Liste zugefügt wird, schon definiert ist !

1.13 Liste aller temporären Assigns

Im Gegensatz zur Liste aller definierten Assigns sind diese, wie der Name schon vermuten läßt, nur temporär, können also nicht unmittelbar gespeichert werden.

1.14 Alle temporären Assigns ausführen

Alle in der temporären Liste definierten Assigns werden nacheinander ausgeführt.

Dabei auftretende Fehler werden abgefangen.

1.15 Markierten temporären Assign ausführen

Es wird nur der markierte Assign ausgeführt.

1.16 Markierten temporären Assign löschen

Der markierte Assign wird ohne Warnung aus der temporären Liste entfernt.

(Siehe auch markierte Definition löschen .)

1.17 Liste der temporären Assigns an Hauptliste anhängen

Die temporäre Liste wird an die Hauptliste angehängt.

1.18 Liste temporärer Assigns löschen

Die temporäre Liste wird gelöscht und das Ausführen-Fenster geschlossen.

1.19 Aktuellen Definition zur Liste hinzufügen

Die Eingaben des Namen , Pfad und Modus werden an die Liste angefügt. Ist der Name bereits in der Liste, so wird eine entsprechende Meldung ausgegeben.

Außerdem wird die Eingabe automatisch in die Liste der temporären Assigns angefügt, jedoch ohne Prüfung auf doppelt vergebene Namen.

1.20 Markierten Definition löschen

Die markierte Definition wird ohne Warnung aus der Liste entfernt.

1.21 Liste speichern

Die Liste wird unter dem aktuellen Name gespeichert. Falls die Liste noch nicht gespeichert wurde, so kann ein Name gewählt werden.

1.22 Liste speichern und Programm beenden

Die Liste wird unter dem aktuellen Name gespeichert und das Programm endet.

1.23 Programm beenden

Das Programm endet ohne das die Liste gespeichert wird. Falls die Liste geändert wurde, wird eine entsprechende Meldung ausgegeben.

1.24 Aktuelle Datei

Hier steht der Name, unter dem die Liste gespeichert wird.

1.25 Das Menü

Die Menü-Leiste enthält zwei Menü-Punkte, die bis auf einige Ausnahmen, die selben Funktionen bereitstellt, wie das Hauptfenster.

1. Projekt:

- * Neu: Die gesamte Liste wird gelöscht. Falls die Liste geändert wurde, wird eine entsprechende Meldung ausgegeben.

- * **Laden:** Eine neue Definitionsdatei kann geladen werden.
Wurde die bestehende Liste geändert, so wird eine entsprechende Meldung ausgegeben.
- * **Anfügen:** Eine Definitionsdatei wird an die bestehende ans Ende angefügt. Sind Namen doppelt vergeben, so wird eine entsprechende Meldung ausgegeben.
- * **Speichern**
- * **Speicher als...:** Wie Speichern, jedoch wird generell ein Name verlangt.
- * **Über:** Eine Copyright-Meldung wird ausgegeben.
- * **Über Projekt:** Informationen über aktuelle Liste.
- * **Über MUI:** Eine Information über MUI wird ausgegeben.
- * **Ende**

2. Optionen:

- * **Icon erzeugen:** Legt fest, ob beim Speichern einer Definitionsdatei ein Icon erzeugt werden soll oder nicht. (Siehe auch ToolTypes.)
- * **MUI:** Startet die Mui -Konfiguration.

1.26 Format der Definitionsdatei

Eine Definitionsdatei kann auch einfach mit einem Editor erstellt werden. Die Datei muß folgendermaßen aussehen:

Am Zeilenanfang wird ein Schlüsselwort erwartet und zwar entweder Locked, Add, Path oder Late. (Siehe Assign-Modus .)

Dann folgt, durch ein "," getrennt, der Assign-Name und durch ein weiteres "," getrennt der zuzuordnende Pfad .

Nun muß die Zeile noch mit einem LineFeed (CHR\$(10)) abgeschossen werden.

Eine Definitionsdatei kann somit auch einfach mittels z.B. "Type" ausgegeben werden.

1.27 Was noch zu tun ist...

???

1.28 ARexx-Unterstützung

```
Befehl      Parameter
=====
AddItem  MODE/A,NAME/A,PATH/A,FORCE/S
```

1.29 ARexx: AddItem

Hiermit wird der Haupt- und der temporären Liste ein Eintrag hinzugefügt. Die Parameter haben folgenden Bedeutung:

MODE=[Locked|Add|Path|Late]

NAME=Name

PATH=Pfad

FORCE Hinzufügen erzwingen.

Rückgabewerte:

RC = 0: Hinzufügen war erfolgreich.

RC = 5: Falls FORCE angegeben war:

Der Name war schon vorhanden, der Eintrag wurde dennoch hinzugefügt.

Sonst:

Der Name war schon vorhanden und der Eintrag wurde nicht hinzugefügt.

RC = 10: MODE war ungültig.

1.30 Über MUI

This application uses

MUI - MagicUserInterface

(c) Copyright 1993/94 by Stefan Stuntz

MUI is a system to generate and maintain graphical user interfaces. With the aid of a preferences program, the user of an application has the ability to customize the outfit according to his personal taste.

MUI is distributed as shareware. To obtain a complete package containing lots of examples and more information about registration please look for a file called "muiXXusr.lha" (XX means the latest version number) on your local bulletin boards or on public domain disks.

If you want to register directly, feel free to send

DM 30.- or US\$ 20.-

to

Stefan Stuntz
Eduard-Spranger-Straße 7
80935 München
GERMANY
