

**WS\_PackHELP**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> WS_PackHELP		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		December 11, 2024	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>WS_PackHELP</b>	<b>1</b>
1.1	WS_PackHELP V1.5 Dokumentation . . . . .	1
1.2	Inhaltsverzeichnis . . . . .	1
1.3	vorwort . . . . .	2
1.4	commodity . . . . .	3
1.5	popup . . . . .	3
1.6	popkey . . . . .	3
1.7	appicon . . . . .	3
1.8	donotwait . . . . .	4
1.9	installation . . . . .	4
1.10	start . . . . .	4
1.11	workbenchstart . . . . .	5
1.12	clistart . . . . .	5
1.13	HauptMenue/Fenster . . . . .	5
1.14	sourcegadget . . . . .	5
1.15	destinationgadget . . . . .	6
1.16	pathgadget . . . . .	6
1.17	stringgadget . . . . .	6
1.18	startgadget . . . . .	6
1.19	optiongadget . . . . .	7
1.20	makedirgadget . . . . .	7
1.21	deletegadget . . . . .	7
1.22	cyclegadget . . . . .	7
1.23	hidegadget . . . . .	7
1.24	aboutgadget . . . . .	8
1.25	quitgadget . . . . .	8
1.26	rungadget . . . . .	8
1.27	newcli . . . . .	8
1.28	load-config . . . . .	8
1.29	save-config . . . . .	9

---

1.30	formatgadget . . . . .	9
1.31	copygadget . . . . .	9
1.32	movegadget . . . . .	9
1.33	helpgadget . . . . .	10
1.34	config . . . . .	10
1.35	packerconfig . . . . .	10
1.36	quellundziel . . . . .	11
1.37	runreq . . . . .	11
1.38	speicher . . . . .	11
1.39	screen . . . . .	12
1.40	screenart . . . . .	12
1.41	aufloesung . . . . .	13
1.42	farben . . . . .	13
1.43	amigaguide . . . . .	13
1.44	aguidefile . . . . .	13
1.45	packerliste . . . . .	14
1.46	packerdir . . . . .	14
1.47	packername . . . . .	14
1.48	packerarchive . . . . .	15
1.49	commandgadget . . . . .	15
1.50	voraussetzungen . . . . .	16
1.51	dateien . . . . .	16
1.52	registrierung . . . . .	17

---

## Chapter 1

# WS\_PackHELP

### 1.1 WS\_PackHELP V1.5 Dokumentation

```
#####          #####  ##   ##   ##   ##   #####  ##          #####
##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##
##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##
#####          #####  #####          #####          #####
##          ##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##
##          ##   ##   ##   ##   ##   ##   ##   ##   ##   ##   ##
##          ##   ##   #####  ##   ##   ##   ##   #####  #####  ##
```

Copyright © 1992, 1993 by Wolfgang Stellwagen

@{ " Inhaltsverzeichnis "    Link Inhalt }

### 1.2 Inhaltsverzeichnis

\*\*\*\*\*

#### Inhaltsverzeichnis

\*\*\*\*\*

- 1.0        Vorwort
- 2.0        Installation
- 3.0        Start von WS\_PackHELP
  - 3.1        Hauptmenue/Fenster
    - 3.1.1      SourceGadget
    - 3.1.2      DestinationGadget
    - 3.1.3      PathGadget
    - 3.1.4      StringGadgets
    - 3.1.5      StartGadget
    - 3.1.6      OptionGadget

- 3.1.7 MakeDirGadget
- 3.1.8 DeleteGadget
- 3.1.9 CycleGadget
- 3.1.10 HideGadget
- 3.1.11 AboutGadget
- 3.1.12 QuitGadget
- 3.1.13 RunGadget
- 3.1.14 NewCli
- 3.1.15 Load-Config
- 3.1.16 Save-Config
- 3.1.17 FormatGadget
- 3.1.18 CopyGadget
- 3.1.19 MoveGadget
- 3.1.20 HelpGadget
- 3.1.21 CommandGadget
- 4.0 PackHELP.Config
- 5.0 Voraussetzungen
- 6.0 Mitgelieferte Programme und Dateien
- 7.0 Registrierung

## 1.3 vorwort

```
*****
1.0      Vorwort
*****
```

Weshalb WS\_PackHELP ?

Es gibt viele Packer wie LHA, ARC, UnZip usw. die man von dem Command Line Interface (CLI) starten und benutzen kann. Sie alle haben aber keine Benutzeroberfläche und können nicht von der Workbench aus gestartet werden. Genau da kommt jetzt WS\_PackHELP ins Spiel. Das Tool unterstützt max. 14 Packer die in der Datei -Packer.config- im Verzeichnis -S:- angegeben werden müssen. Wie die Angaben in der Configdatei aussehen müssen kannst du nachlesen in der Datei WS\_PackHELP.config.dok, in dieser Dokumentationsdatei sind alle Beschreibungen enthalten die, die Configdatei betrifft. Also wie gesagt mit WS\_PackHELP können nun diese Packer auch über die WB benutzt werden. Allerdings müssen die Packer auch auf deiner Festplatte oder auf der Diskette sein mit der du arbeitest, denn das Programm ruft alle diese Packer aus den angegebenen Verzeichnissen auf.

Alle Packer die in der mitgelieferten Configdatei aufgeführt sind habe ich getestet und das Programm arbeitet einbandfrei mit diesen Packern. WS\_PackHELP unterstützt alle Optionen die in den einzelnen Packer von Ihren Programmierern eingebaut worden sind.

WS\_PackHELP ist als Commodity geschrieben, und unterstützt einige Parameter die bei einem Start berücksichtigt werden.

## 1.4 commodity

Commodity

-----

Folgende Parameter werden von dem PRG WS\_PackHELP unterstützt.

CX\_POPUP            CX\_POPKEY

CX\_APPICON        DONOTWAIT

## 1.5 popup

CX\_POPUP    => Mit diesem Parameter kannst du angeben ob bei Start von der WB oder CLI das Fenster von WS\_PackHELP geöffnet wird oder ob das PRG auf die Eingabe von einer Tastenkombination warten soll bis es aktiviert wird.  
(Default: YES)

## 1.6 popkey

CX\_POPKEY   => Hier wird die Tastenkombination angegeben wenn das PRG sein Fenster öffnen soll, falls du CX\_POPUP=NO gewählt hast.  
Ist die WB nicht geladen und das Gadget -HIDE- im PRG wird ausgewählt schließt das PRG sein Fenster und wartet im Hintergrund auf die Tastenkombination die hier eingestellt wurde.  
Falls die WB aber geladen ist und das Gadget -HIDE- wird ausgewählt und CX\_APPICON=YES eingestellt wurde schließt das PRG zwar sein Fenster, platziert aber ein AppIcon auf der WB und dann kann das PRG-Fenster nur durch einen Doppelklick auf das Piktogramm geöffnet werden. Ist aber CX\_APPICON=NO eingestellt funktioniert das Ganze analog zu dem ersten Verfahren (Warten auf eine Tastenkombination).  
(Default: shift numericpad \*)

## 1.7 appicon

CX\_APPICON => Dieser Parameter ist eine Besonderheit des Programmes, d.h. ist CX\_APPICON=YES eingestellt und ist die Workbench geladen, wird bei dem Start von WS\_PackHELP ein AppIcon auf der WB platziert. Durch einen Doppelklick auf das Piktogramm wird das Programm aktiviert, d.h. WS\_PackHELP öffnet sein Hauptfenster und man kann wieder mit dem Programm arbeiten.  
Wenn die Workbench geladen ist und das AppIcon auf der Workbench platziert wurde kann man Gleichzeitig mit diesem AppIcon dann Packerarchive die ein Programmpiktogramm haben darüber ziehen und das Archiv wird nach Auswahl eines Verzeichnisses automatisch dorthin entpackt.

(Default:YES)  
 Hinweis:  
 Als Packer wird der Packer verwendet der momentan im  
 geöffneten Fenster aktiv ist (Siehe Gagdet Packer).  
 In der momentanen Version werden nur Archive entpackt.

W I C H T I G  
 =====

Falls CX\_APPICON=YES eingestellt wurde, muss CX\_POPUP=NO  
 gesetzt sein wenn das PRG ein AppIcon auf der Workbench  
 beim Start plazieren soll, andernfalls hat CX\_POPUP=YES  
 Vorrang vor CX\_APPICON=YES und das PRG öffnet sein Fenster.

## 1.8 donotwait

DONOTWAIT => Bei einem Start des Programmes aus der Schublade -WBStartUp-  
 ist es wichtig diesen Parameter in dem Programmpiktogramm in  
 Tool Types mit anzugeben da sonst ein Requester auf der WB  
 erscheint und auf eine Rückmeldung von dem Programm wartet.

## 1.9 installation

\*\*\*\*\*  
 2.0 Installation  
 \*\*\*\*\*

Um das Programm zu installieren muß das Programm Install gestartet  
 werden.

Der Aufruf sieht folgendermaßen aus:

```
CD Device/Dir
Execute Install Install-WS_PackHELP
```

oder ab Kick 1.3

```
Install Install-WS_PackHELP
```

Device/Dir = Das Device oder das Verzeichnis in dem sich das Programm  
 und die mitgelieferten Dateien befindet.

## 1.10 start

\*\*\*\*\*  
 3.0 Start von WS\_PackHELP  
 \*\*\*\*\*

Von der Workbench oder von dem CLI



## 1.11 workbenchstart

### 3.0.1 Workbenchstart

-----

Durch einen einfachen Doppelklick auf das Piktogramm wird WS\_PackHELP gestartet.  
Da das Programm als Commodity geschrieben ist kann über das Info-Piktogramm von WS\_PackHELP bei ToolTypes dann folgende Werte eingestellt werden:  
z.B.:  
CX\_POPUP=YES  
  
CX\_POPKEY="control p"  
  
CX\_APPICON=NO

## 1.12 clistart

### 3.0.2 Clistart

-----

Im CLI sollte man am besten WS\_PackHELP mit dem Befehl RUN starten.  
Da das Programm als Commodity geschrieben ist kann man über die Parameter verschiedene Werte einstellen die beim Start des PRG's berücksichtigt werden.  
z.B.:  
RUN <Pfad> <PRGName> [CX\_POPUP/K,CX\_POPKEY/K,CX\_APPICON/K]  
  
RUN SYS:WS\_PackHELP CX\_POPUP=YES CX\_POPKEY="control p" [RETURN]

## 1.13 HauptMenue/Fenster

\*\*\*\*\*  
3.1 HauptMenue/Fenster  
\*\*\*\*\*

Beschreibung aller im Programm verwendete Gadgets

## 1.14 sourcegadget

### 3.1.1 SourceGadget

-----

Hier wird das Verzeichnis ausgewählt, daß deine Archive zum Entpacken oder eine/mehere Datei(en) zum Packen enthält. Wird das Gadget angeklickt öffnet sich automatisch ein Datei-Requester (Multiselect) mit dem du dann Dateien auswählen kannst.

## 1.15 destinationgadget

### 3.1.2 DestinationGadget

---

Es wird ein Verzeichnis-Requester geöffnet und durch die Auswahl eines Verzeichnisses werden die Archive dann dorthin entpackt oder beim Packen wird das angegebene Archive dort abgelegt.

## 1.16 pathgadget

### 3.1.3 PathGadget

---

Falls der Pfad von dem Pfad in der Configdatei abweicht ist es hier möglich ein Verzeichnis auszuwählen in dem sich der Packer befindet mit dem man arbeiten möchte.

## 1.17 stringgadget

### 3.1.4 StringGadgets

---

Als Alternative zu den Auswahlrequestern kannst du auch hier die Quell und Zielverzeichnisse oder den Pfad zu dem jeweiligen Packer eingeben. Sie zeigen auch immer die aktuellen Verzeichnisse an. Eine Besonderheit ist bei dem SourceStringGadget zu beachten, hier kan man auch Dateinamen per Hand eingeben die dann verarbeitet werden.

Es ist auch möglich mehrere Dateien einzugeben, allerdings müssen diese Dateinamen durch ein Levrzeichen (Space) getrennt werden.

Nur dann ist es auch sichergestellt, das die Dateien verarbeitet werden, unter Berücksichtigung des momentan eingestellten Befehl.  
{@" (Siehe CommandGadget) " Link CommandGadget}

## 1.18 startgadget

### 3.1.5 StartGadget

---

Sobald dieses Gadget ausgewählt wird, wird vor dem Ausführen der eingestellten Parameter noch ein Fenster geöffnet, in dem dann die voreinstellten Werte von dem Optiongadget noch verändert werden kann, d.h. das Programm gibt dir die Möglichkeit noch vor der Ausführung einige Parameter zu ändern. Dieses Fenster hat noch eine Zusatzfunktion die wichtig bei dem Packen von Daten ist. Beim Packen werden auf dem Fenster zusätzlich zwei Gadgets angezeigt mit denen es dann Möglich ist ein altes bestehendes Archiv oder ein neues Archive auszuwählen. Erst bei einem Klick auf das OK-Gadget wird der eingestellte Packer aufgerufen, alle ausgewählten Parameter werden mit gegeben und das Programm führt die Anweisungen aus.

## 1.19 optiongadget

### 3.1.6 OptionGadget

---

Nach dem Anwählen dieses Gadgets wird ein Preferences-Fenster geöffnet in dem man dann einige Einstellungen vornehmen kann. Das Fenster dient zum Einstellen von Farbe, Auflösung, Breite und Höhe des Programm-Screens und/oder Windows. Zeigt je nach Auswahl auch noch den verfügbaren Speicher oder das aktuelle Datum und Zeit an. über das Gadget Packer können dann die Voreinstellungen für die einzelnen Packer gemacht werden. Alle Voreinstellungen werden beim Sichern der Configurationdatei mit abgespeichert.

## 1.20 makedirgadget

### 3.1.7 MakeDirGadget

---

Bei dem Umgang mit Archivierungsprogrammen ist es manchmal angebracht Verzeichnisse zu erstellen, in das dann die Daten entpackt werden oder wenn beim Packen das Archiv dann dort abgelegt werden soll. Dieses Gadget oeffnet nach Auswahl ein Fenster in dem man dann über das StringGadget einen neuen Verzeichnisnamen eingeben kann und erstellt das Verzeichnis.

## 1.21 deletegadget

### 3.1.8 DeleteGadget

---

Hier kannst du einzelne Dateien löschen oder Verzeichnisse die keine Unterverzeichnis haben. Soll ein Verzeichnis gelöscht werden in dem Unterverzeichnisse enthalten sind werden nur die Dateien gelöscht. Zur Zeit ist es leider noch nicht möglich Dateien oder Verzeichnisse rekursiv zu löschen (Aber in Arbeit).

## 1.22 cyclegadget

### 3.1.9 CycleGadget

---

Über dieses Gadget wird der Packer eingestellt mit dem man Arbeiten möchte.

## 1.23 hidegadget

### 3.1.10 HideGadget

-----  
Ist die Workbench gestartet worden wird ein AppIcon auf der Workbench platziert und das Hauptfenster wird geschlossen. Sollte die Workbench noch nicht geladen worden sein wird das Hauptfenster geschlossen und das Programm wartet im Hintergrund auf eine Tastenkombination (Siehe Commodity -CX\_POPKEY- ) um das Arbeitsfenster wieder zu oeffnen. ↔

## 1.24 aboutgadget

### 3.1.11 AboutGadget

-----  
Es wird eine Copyrightmeldung und auch die Versionsnummer von WS\_PackHELP angezeigt.

## 1.25 quitgadget

### 3.1.12 QuitGadet

-----  
Das Programm wird über diesen Schalter wenn der Requester mit OK beantwortet wurde beendet.

## 1.26 rungadget

### 3.1.13 RunGadget

-----  
Hier kannst du jedes Programm starten um damit zu arbeiten.  
Alle Programme werden asynchron gestartet, also unabhaengig von WS\_PackHELP.

## 1.27 newcli

### 3.1.14 NewCli

-----  
Öffnet eine neue Shell oder CLI mit der man dann arbeiten kann.

## 1.28 load-config

### 3.1.15 Load-Config

-----  
Nach dem Öffnen eines Requesters und Auswahl der Configuration-datei werden alle vorher benutzten Voreinstellungen gelöscht und

---

WS\_PackHELP arbeitet mit den voreingestellten Werte aus der Configdatei weiter.

Hinweis: Sollte die Configdatei beim Laden aus irgenwelchen Gründen nicht vorhanden sein setzt WS\_PackHELP Defaultwerte, die dann durch ändern der Preferences nach Deinen eigenen Anforderungen verändert werden können.

## 1.29 save-config

### 3.1.16 Save-Config

-----

Die aktuellen Daten z.B. Auflösung, Farbe Packervoreinstellungen etc. werden in der Datei -S:Packer.config- gespeichert, falls kein anderer Name über den Auswahlrequester ausgewählt wurde.

Hinweis: Alle Daten/Werte im Preferences und -PackerFenster werden beim Sichern abgelegt in die Configdatei.

## 1.30 formatgadget

### 3.1.17 FormatGadget

-----

Über diesen Menuepunkt kannst du Disketten formatieren. Es wird ein Window geöffnet mit dem du dann die Einstellung wählen kannst mit der dann die Diskette formatiert wird.  
Es wird das normale OFS und das FFS unterstützt.

## 1.31 copygadget

### 3.1.18 CopyGadget

-----

Hier kannst du Dateien mit dem Multiselectrequester auswählen und in ein neues Verzeichnis kopieren.

Zuerst wird nach dem Verzeichnis gefragt wohin die Dateien kopiert werden sollen und danach müssen die einzelnen Dateien ausgewählt werden die kopiert werden sollen.

## 1.32 movegadget

### 3.1.19 MoveGadget

-----

Hier ist der Vorgang analog zum CopyGadget nur mit dem Unterschied das die ausgewählten Dateien verschoben werden.

---

## 1.33 helpgadget

### 3.1.20 HelpGadget

-----  
Die Dokumentation wird im AmigaGuide-Format angezeigt.  
Die Online-Hilfe läuft parallel zum Programm und ist während  
des Arbeiten beim Einstieg in das Programm ganz Nützlich.

## 1.34 config

```
*****  
*                4.0          Configuration                *  
*****
```

Anzeige für Speicher, Datum und Zeit

Art des aktiven Screens setzten

Breite und Höhe des Screen setzten

Auflösung des Screen bestimmen

Farben des Screen bestimmen

Packereinstellungen

## 1.35 packerconfig

```
*****  
* Verzeichnisse, Packeroptionen , Packerpattern für die benutzen *  
* Packer setzen.                *  
*****
```

PackerVerzeichnis

Quell und Zielverzeichnisse

Verzeichnis für Runrequester

Verzeichnis für AmigaGuide

Verzeichnis für das AmigaGuide-File

PackerListe

PackerName

PackerArchive

---

## 1.36 quellundziel

```
*****
*          Pfad für Quell- und Zielverzeichnis setzen          *
*****
```

Die Verzeichnisse die hier eingestellt wurden, werden im Hauptfenster als Defaultverzeichnisse angezeigt, d.h. beim öffnen des Source oder Destination-Requesters wird zuerst immer das jeweilige Verzeichnis eingelesen, daß hier eingegeben wurde.

Beispiel:

Die zu entpackenden Dateien befinden sich auf dem Device -SYS- und im Unterverzeichnis -Downloads-.

Als Zielverzeichnis also das Verzeichnis in daß, das Archive entpackt oder gepackt werden soll befindet sich auch auf dem Device -SYS- aber im Unterverzeichnis -Sonstiges-.

Im Hauptfenster erscheinen die Verzeichnisse im SourceStringGadget

SYS:Downloads

und im DestinationStringGadget

SYS:Sonstiges

DEFAULT: SYS:

## 1.37 runreq

```
*****
*          Pfad für Run-Requester setzen          *
*****
```

Hier wird ein Verzeichnis angegeben, das beim Öffnen des RunRequesters zuerst das Verzeichnis einliest, das durch deine Voreinstellung angezeigt wird.

Selbstverständlich kann man auch andere Verzeichnisse auswählen um dann von dort ein Programm zu starten. Die gestarteten Programme laufen dann als eigenständige Tasks.

DEFAULT: SYS:

## 1.38 speicher

```
*****
*          Anzeige für Speicher, Zeit, Datum setzen          *
*****
```

Zum Anzeigen von Memory, Datum und Zeit muß jedes der drei Gadgets mit

einem Haken versehen sein, erst dann werden im Hauptfenster die Werte angezeigt.

Falls du den aktuellen Speicher, d.h. deinen noch vorhandenen Speicher nicht anzeigen lassen willst mußt du durch einen Klick auf das Gadget den angezeigten Haken löschen.

```
DEFAULT: Memory-Anzeige = An
         Datums-Anzeige  = An
         Zeit-Anzeige    = An
```

## 1.39 screen

```
*****
*          Screenbreite und Höhe / Windowbreite und Höhe setzen          *
*****
```

Um die Breite und Höhe eines Screen einzustellen auf dem man Arbeiten will, muß man den Parametern Screen-Width, Screen-Height, Window-Width und Window-Height Werte zuweisen.

Das ganze ist von Vorteil wenn man mit einem Multisynchron-Monitor arbeitet und die Frequenz von 50 Hz auf 60 Hz erhöht hat.

```
DEFAULT: ScreenMode   = HIRES
         Screen-Width = 640
         Screen-Height= 256
         Window-Width = 256
         Window-Height= 640
```

```
DEFAULT: ScreenMode   = HIRES INTERLACE
         Screen-Width = 640
         Screen-Height= 512
         Window-Width = 512
         Window-Height= 640
```

## 1.40 screenart

```
*****
*          Art des Screen bestimmen          *
*****
```

Es gibt insgesamt drei Möglichkeiten einen Screen auf dem man arbeiten will einzustellen:

1. WorkBench
2. Custom
3. Public

1. Alle Fenster werden auf der Workbenchscreen geöffnet, alle eingestellten Werte werden mit übernommen.

2. Alle Fenster werden auf einem eigenen Screen geöffnet, der dann die eingestellten Farben, Auflösung usw. übernimmt.



3. Die Fenster werden auf der einem Publicscreen geöffnet, die eingestellten Werte werden mit übernommen.

## 1.41 aufloesung

```
*****
*           Auswahl von Auflösung  des Fensters           *
*****
```

Das Programm unterstützt in dieser Version nur die Auflösungen Hires und Hires Interlaced. Die maximalen Werte bei den beiden Screen-auflösungen sind bei:

```
HIRES          = 640 x 256
HIRES INTERLACED = 640 x 512
```

DEFAULT: HIRES

## 1.42 farben

```
*****
*           Auswahl der Farben eines Screens und Fensters   *
*****
```

Das Programm unterstützt nur 2 Bitplanes also nur 4 Farben die eingestellt werden können.

```
DEFAULT: Color0=aaa,
         Color1=000,
         Color2=fff,
         Color3=68b
```

## 1.43 amigaguide

Gibt das Verzeichnis an in dem das AmigaGuide-Programm abgelegt wurde. Das Ganze ist von Vorteil wenn man sich nicht an die von Commodore vorgegebenen Verzeichnisse hält. (äh, ich zum Beispiel und einige Bekannte, die ich kenne).

Außerdem bleibt WS\_PackHELP im Programmcode dann wesentlich kürzer wenn man die Programme extern aufruft als wenn man das Ganze in das Programm einfügt.

## 1.44 aguidefile

Gibt das Verzeichnis an in dem die Datei WS\_PackHELP.guide abgelegt wurde. Es ist dann einfacher darauf zuzugreifen als wenn es in einem von mir vorgegebenen Verzeichnis liegt. (ich zum Beispiel habe alle Dokumentationen in einem separaten Verzeichnis -DOC- und wenn es Programme zulassen sind meine Guidedateien auch in einem separaten Verzeichnis, d.h. alles auf einen Blick.

## 1.45 packerliste

```
*****
*                PackerListe                *
*****
```

Hier werden die Packer angezeigt, mit denen du arbeiten willst und die von dir in diese Liste eingefügt wurden. Es sind max. 14 Packer möglich, die das Programm verwalten kann. Wie man Packer in diese Liste einfügen oder löschen kann, ist unter dem Punkt PackerName zu finden

## 1.46 packerdir

Hier muß für jeden Packer den man benutzen will das Verzeichnis angeben in dem der Packer zu finden ist.

Beispiel:

Der Packer DMS befindet sich in dem Verzeichnis SYS:Packer/DMS, dann muß damit WS\_PackHELP auch richtig arbeitet das Verzeichnis ausgewählt werden in dem der Packer zu finden ist.

## 1.47 packername

Das Hinzufügen von Packern in die Liste ist nur durch dieses StringGadget möglich.

Als erstes muß der Name des Packers genauso eingegeben werden wie er auch Wirklichkeit heißt, es sind aber nur max 9 Zeichen möglich. Danach muss das ButtonGadget -ADD- ausgewählt werden und der Packer wird in die Liste aufgenommen. Natürlich kann ein Packer auch wieder aus der Liste entfernt werden, das wiederum besorgt das ButtonGadget -DEL-. Aber Vorsicht es wird nur der Packer entfernt der in der Liste selektiert wurde.

Hinweis:

Beim erstmaligen installieren der Packer empfehle ich zuerst die Liste zu füllen und dann erst nach selektieren des Packers das Packerverzeichnis und die Packerarchivefunktionen einzustellen.

## W I C H T I G

-----

Sollte jemand den Packer DMS benutzen dann muss der Packername von DMS auch DMS (Großschrift) heißen und nicht DMS1.11, denn so wie der Name eingegeben wurde wird er auch im Hauptfenster angezeigt.

Warum wird man sich fragen soll bei dem Packer DMS in Großbuchstaben der Name DMS eingegeben werden, nun dieser Packer arbeitet etwas anders und um alle Funktionen nutzen zu können habe ich einige Konstanten eingebaut, damit der Packer auch einwandfrei arbeitet.

## 1.48 packerarchive

WS\_PackHELP unterstützt alle Optionen ein eingestellten Packers, aber damit WS\_PackHELP richtig bei jedem Packer arbeitet muß man für jeden Packer noch einige Archivefunktionen des jeweiligen Packers einstellen.

## W I C H T I G

-----

Vorab noch eine Information, alle Funktionen die von den einzelnen Packern nicht unterstützt werden müssen durch ein -NONE- in den dafür vorgesehenen StringGadgets und in Großschrift eingegeben werden.

1. Create
2. Extract
3. List
4. Test
5. Option
6. Pattern

1. In diesem StringGadget muß der Packbefehl des selektierten Packers eingetragen werden.
2. Bei diesem StringGadget ist der Entpackbefehl des Packers einzutragen.
3. Der Name -List- sagt uns schon wie diese Eingabe lauten muss, es ist der List oder -Viewbefehl des Packers einzutragen.
4. Der Testbefehl des Packers ist hier einzutragen.
5. Und Optional kann man hier einige Optionen des eingestellten Packers eintragen.  
(Einzige Ausnahme bei dem WS\_PackHELP-Kommandobefehl SPECIAL
6. Mit Pattern sind die Endungen (.lha, .dms, .arc usw) des verwendeten Packers einzutragen.

## 1.49 commandgadget

WS\_PackHELP unterstützt alle Kommandos und Optionen eines eingestellten Packers, diese Optionen werden durch das KommandoGadget links von dem CycleGadget der Packer eingestellt. Im folgenden werde ich alle Kommandos die unterstützt werden beschreiben.

1. Entpack
2. Pack
3. View
4. Test
5. Special
6. Help

1. Entpackt alle Dateien die durch den Multiselect-Requester ausgewählt wurden in das voreingestellte Zielverzeichnis.
2. Das Packen von Dateien erfolgt analog zu dem Entpacken von Archiven.
3. Mit diesem Kommando ist es möglich ein Archiv das gepackt wurde aufzulisten um zu sehen welche Dateien sich in dem Archiv befinden.
4. Dieses Kommando testet das ausgewählte Archiv auf irgendwelche Fehler z.B. I/O-ERROR usw.
5. Eine Besonderheit ist dieses Kommando, hier werden alle voreingestellten Werte von dem jeweiligen PackerArchiv nicht berücksichtigt, d.h. es sind nur Befehle von Bedeutung die im OptionStringGadget bei Start des Packers eingestellt wurden.  
Erst hier ist es möglich alle Kommandos eines Packers zu nutzen.  
Das allerdings setzt einige Kenntnisse von dem verwendeten Packer voraus.  
Also wenn dieses Kommando eingestellt ist sind nur die Kommandos und Optionen aktiv die im OptionGadget eingegeben wurden.
6. Durch den Aufruf des eingestellten Packers ohne Parameter wird der Hilfetext des jeweiligen Packers angezeigt.

## 1.50 voraussetzungen

```
*****
*                5.0      Voraussetzungen zum Betrieb      *
*****
```

Das Programm WS\_PackHELP ist nur unter Kickstart 2.0 lauffähig, aber eine Version die auch unter Kickstart 1.2/1.3 laufen kann ist in Arbeit.

## 1.51 dateien

```
*****
*                6.0      Mitgelieferte Verzeichnisse und Dateien      *
*****
```

\*\*\*\*\*

Install  
Install.info  
Installer  
Install-WS\_PackHELP  
WS\_PackHELP  
WS\_PackHELP.info  
ReadMe  
ReadMe.info

S/Packer.config

Docs/Deutsch/WS\_PackHELP.dok  
Docs/Deutsch/WS\_PackHELP.dok.info  
Docs/Deutsch/WS\_PackHELP.guide  
Docs/Deutsch/WS\_PackHELP.guide.info

Docs/English/WS\_PackHELP.doc  
Docs/English/WS\_PackHELP.doc.info  
Docs/Englich/WS\_PackHELP.guide  
Docs/Endlich/WS\_PackHELP.guide.info

Registration/Deutsch/Print\_Registrierung\_Form  
Registration/Deutsch/Print\_Registrierung\_Form.info  
Registration/Deutsch/Registrierung\_Form  
Registration/Deutsch/Registrierung\_Form.info

Registration/English/Print\_Registration\_Form  
Registration/English/Print\_Registration\_Form.info  
Registration/English/Registration\_Form  
Registration/English/Registration\_Form.info

## 1.52 registrierung

### WS\_PackHELPV2.0 Registrierung

~~~~~

Falls jemand der mit dem Programm arbeitet sich entscheiden kann, ob er/sie sich bei mir als Shareware-User registrieren lassen will und somit alle neueren Versionen des Programmes erhält, der füllt bitte das mitgelieferte Registrierformular aus und sendet es dann an meine Adresse.

Das Formular befindet sich im Verzeichnis Registrierung und dort ist dann auch meine Adresse und die Höhe der Sharewaregebühr zu finden.  
Im voraus vielen Dank.

WOLFGANG