

HyperHelp

Helmut Lutzenberger

COLLABORATORS

	<i>TITLE :</i> HyperHelp		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Helmut Lutzenberger	November 24, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	HyperHelp	1
1.1	HyperHelp	1
1.2	Das Versionsdurcheinander	2
1.3	Willkommen zu HyperHelp	2
1.4	Wie kriegt man das Teil zum laufen	4
1.5	So passt sich HyperHelp an ihren Geschmack an	5
1.6	Der Voreinsteller	8
1.7	Index, NEIN kein Index nur die Erklärung zum Index-Gadget	9
1.8	Die Eingabeüberwachung, Watchdog	9
1.9	priority	10
1.10	buffer	10
1.11	Stream Buffer	10
1.12	HotKey	11
1.13	Force Key	11
1.14	Editor	11
1.15	Port	11
1.16	Command	11
1.17	Save	12
1.18	Quit	12
1.19	Build Index	12
1.20	The index file	12
1.21	Das Arbeitstier	14
1.22	Rights and reservations	15
1.23	The CLI index creator	16
1.24	The CLI preferences program	17
1.25	Die Programmierer	17
1.26	Bugs	18
1.27	About Magic User Interface	19
1.28	Dankeschön	20
1.29	You did WHAT?!?!?	20

Chapter 1

HyperHelp

1.1 HyperHelp

HyperHelp

v1.00

©1994 Apocalypse Productions

Sean Russell

Helmut Lutzenberger

!!Die deutsche Anleitung ist leider noch nicht ganz fertig, deshalb!!

!!sind einige Texte noch in englisch.!!

Inhalt

Version

Einleitung Übersicht und Systemanforderungen.

Installation

Konfiguration Wichtige Grundeinstellungen.

HyperHelp Der Voreinsteller.

HyperHelp_cx Das Commodity.

MakeIndex Der CLI Index-generator.

MakePrefs Der CLI Preferences-generator.

Copying Copyright und rechtliches

Autoren oder, wie erreiche ich dir Programmierer

Bugs und die Zukunft von HyperHelp

MUI Das geniale GUI.

Danksagungen

Commodore :- (von Sean Russell)

1.2 Das Versionsdurcheinander

Da HyperHelp nicht ein Programm alleine ist sondern ein Packet aus mehreren Programmen und jedes Programm seine eigene kleine Geschichte besitzt, gibt es im Grunde keine einheitliche Versionsnummer. Kurz es herrscht ein riesiges Versionsdurcheinander.

Diesem Chaos haben wir wie folgt Einhalt geboten:

- * Jedes Programm besitzt in seinem Versionsstring eine Revisionsnummer.

Diese Nummer ist nur wichtig, falls ein spezielles Programm bei Ihnen zu Systemabstürzen führt. Ansonsten hat diese Revisionsnummer nur für Sean und mich eine Bedeutung.

- * Das ganze Packet besitzt eine Release-Nummer. Diese ist kennzeichnet das ganze Archiv und sollte bei allen allgemeinen Fragen zu HyperHelp angegeben werden.

An ihr wird man auch zukünftige Updates erkennen.

- * Und dann gibt es noch die Versionsnummer dieses Textes, den sie gerade lesen und die ist immer gleich der Release-Nummer.

Kurz die Versionsnummer von HyperHelp sind genauso aufgebaut wie die des Amiga Betriebssystems. z.B. WB 2.1 entspricht intern der Revision 37.xx.

1.3 Willkommen zu HyperHelp

HyperHelp ist ein Packet zur Einrichtung und Wartung eines auf AmigaGuide(TM) basierende Online-Hilfssysteme.

Nichts Neues wird sich jetzt so mancher denken. Richtig! Aber

- * HyperHelp ist so gut wie umsonst. Jeder kann es ohne schlechtes Gewissen auf seinem Amiga installieren. Was natürlich nicht heissen soll das einer der beiden Programmieren Geldspenden zurückschicken wird.

- * HyperHelp unterstützt nicht nur AmigaGuide(TM), vielmehr ist es komplett auf AmigaGuide aufgebaut. D.h. jeder AmigaGuide-Text kann einfach mittels 'Drag-And-Drop' in das HyperHelp-System integriert werden.

- * Ein absolut geniales MUI-Preferences Programm.

- * HyperHelp ist asynchron. Das heißt jeder Hilfstext wird als eigener Prozess gestartet, es können folglich beliebig viele Hilfstexte gleichzeitig geöffnet sein, bis der Speicher platzt.

- * Intelligente Hotkey-Auswertung. HyperHelp erkennt die Umgebung in der Sie gerade arbeiten und verwendet entsprechend verschiedene Methoden um sich aus dieser Umgebung den Suchbegriff zu holen.

- * Einzigartige Eingabestrom-Überwachung. Im Gegensatz zu seinen

Konkurrenten ist HyperHelp in der Lage auch dann noch ein Schlüsselwort aus dem Eingabestrom zu ziehen falls dieser editiert wurde.

* HyperHelp verwendet zur Suche den Levenshtein-Algorithmus.

Dieser geniale Algorithmus erkennt nicht nur exakt gleiche Suchbegriffe sonder auch ähnliche.

* HyperHelp unterstützt pattern-matching. Suchbegriffe dürfen die Wildcards '?' und '*' enthalten.

* HyperHelp ist furchtbar bescheiden und verbraucht daher nur wenig Speicher und noch weniger CPU-Zeit.

* Natürlich liefern wir ein Installerskript mit.

* Und vor allem HyperHelp ist International. Den Apocalypse Productions hat 'Niederlassungen' in den USA sowie in Europa.

* Und HyperHelp ist recyclebar, will sagen jeder der immer noch nicht überzeugt ist daß er HyperHelp unbedingt haben will kann es jetzt ohne bedenken von seiner Festplatte löschen.

HyperHelp besteht im wesentlichen aus folgenden vier Programmen:

- **HyperHelp** - Das Preferences Programm.

- **HyperHelp_cx** - Das Commodity.

- **MakeIndex** - Ein Shell-Programm mit dem man die von HyperHelp_cx benötigten **Index**-Dateien erzeugen kann.

- **MakePrefs** - Ein Shell-Programm mit dem man die Voreinstellungen für HyperHelp setzen kann.

MakeIndex und MakePrefs sind vor allem für alle die gedacht, die nicht eine registrierte Version von **MUI** auf ihrem Amiga installiert haben.

Damit HyperHelp_cx arbeiten kann werden von HyperHelp noch diverse Index und eine Preferences-Datei erzeugt. Für die Hilfstexte sind sie selbst verantwortlich. Besonders nützlich ist HyperHelp in Verbindung mit den AutoDocs(TM) von Commodore (können z.B. über Hirsch&Wolf bezogen werden). Natürlich kann man mit HyperHelp auch Hypertexte mit Kochrezepten oder FTP-Servern oder benutzen, man muß sie sich nur schreiben oder z.B. aus dem AmiNet saugen.

Eine vollständige Liste mit den Dateien welche im HyperHelp Archiv enthalten sein sollten ist exe.lst.

HyperHelp benötigt OS2.0 (V37+) und natürlich die amigaguide.library V34+.

Das MUI-Preferences Programm benötigt zusätzlich die muimaster.library v4 oder höher. Die **muimaster.library** ist nicht im Archiv

enthalten, ist aber frei erhältlich. Genauer Informationen sind im Kapitel

MUI

Verzichtet man auf den Voreinsteller mit graphischer Benutzeroberfläche und

benutzt die beiden Programm **MakePrefs** und **MakeIndex** so ist das

HyperHelp-Packet auch ohne MUI voll funktionsfähig.

HyperHelp_cx wartet im Hintergrund bis Sie eine der beiden Hotkey Kombinationen 'ForceKey' (voreingestellt rechte Alt-Taste und Help) oder 'Hotkey' (voreingestellt rechte Shift-Taste und Help) drücken. Bei erster Kombination versucht HyperHelp_cx nicht aus der momentanen Umgebung ein Schlüsselwort zu ziehen, sonder öffnet eine Übersicht mit allen Hilfsdateien auf die HyperHelp zugriff hat. Bei der Kombination 'Hotkey' versucht HyperHelp_cx die Umgebung in der Sie gerade arbeiten zu erkennen und aus dieser ein Schlüsselwort zu ziehen. Danach durchsucht HyperHelp seine **Index**-Datei nach diesem Schlüsselwort.

1.4 Wie kriegt man das Teil zum laufen

- Übersicht

Drei Dinge sind zu erledigen, damit das HyperHelp auch auf Ihrem Amiga seinen Dinst versieht:

- 1 - Pfad und Name der **Index**-Datei setzten.
- 2 - Den ARexx-Port und den entsprechende ARexx-Befehl ihres Editors einstellen.
- 3 - Die **Index**-Datei erzeugen.

- Installer

Um es ihnen so leicht wie möglich zu machen hat Sean ein Installerskript geschrieben, welches das komplette HyperHelp System auf ihrer Festplatte installiert. Sie benötigen dazu nur den 'Installer' von Commodore. Sollten sie diesen noch nicht besitzen ist das nicht weiter tragisch, da er z.B auf den Fishs-Disks oder im AmiNet zu finden ist.

Und für alle die Maschinen grundsätzlich nicht trauen und deshalb HyperHelp lieber zu Fuß installieren möchten, hier die Funktionalität des Installer-skripts:

- a) Kopiere **HyperHelp** (und Icon) nach prefs/
- b) Kopiere **HyperHelp_cx** (und Icon) nach WBStartup/
- c) Kopiere **MakeIndex** und **MakePrefs**

nach C: (Diese Schritt können sie sich sparen falls sie eine registrierte Version on MUI besitzen)

d) und zu zum Schluß einfach noch diese Anleitung in ihr bevorzugtes Hilfstextverzeichnis kopieren und das HyperHelp System ist installiert.

Aber nachdem sich Sean so viel Mühe mit dem Installerskript gemacht möchte ich an dieser Stelle jeden bitte es auch zu benutzen.

1.5 So passt sich HyperHelp an ihren Geschmack an

Im Gegensatz zur Installation erfordert die Konfiguration doch ein wenig Arbeit. Alle registrierten Benutzer von MUI haben es am einfachsten. Sie können alle Grundeinstellungen über das Preferences Programm **HyperHelp** vornehmen. Ansonsten müssen die Grundeinstellungen über die Tooltypes von gesetzt werden.

HyperHelp_cx benutzt als erstes die Einstellungen in der Konfigurationsdatei (in envarc:), diese wird mit HyperHelp erzeugt. Ist keine Konfigurationsdatei vorhanden verwendet HyperHelp_cx die in den Tooltypes eingestellten Werte, sind auch diese nicht vorhanden verwendet es seine internen Grundeinstellungen.

HyperHelp_cx erkennt die folgenden Tooltypes:

* port

Dies ist der Name des ARexx-Ports ihres Editors. Über diesen kommuniziert HyperHelp_cx mit ihrem Editor. Falls ihr Editor eine ARexx-Schnittstelle besitzt so finden sie den Portnamen in der Dokumentation ihres Editors.

Beispiele:

- rexx_ced bei CygnusEd
- GOLDED.1 bei GoldEd.

* command

Dies ist das ARexx-Kommando das HyperHelp_cx ihrem Editor schicken muß um von diesem das Wort unter dem Cursor zu erhalten. Dieses sollte in der Anleitung des Editors stehen.

Beispiele:

- GETWORD bei CygnusEd und Edward
- GETWORD, EXTRACT bei GoldEd

* index

Dies ist der ganze Pfad und der Dateiname der **Index**-Datei

* cx_priority

Die Priorität von HyperHelp_cx. Voreingestellt ist 0.

* buffer

Die Größe des E/A - Puffers. Je größer dieser gewählt wird umso schneller ist HyperHelp_cx bei der Suche nach Schlüsselwörtern. Mehr als 10000 ist möglich im Grunde aber Speicherverschwendung. Voreingestellt ist 10000.

* hotkey

Die Tastenkombination mit der HyperHelp_cx angestoßen wird einen Hilfstext auf den Bildschirm zu bringen. Voreingestellt ist 'lshift

help', also die linke Shifttaste zusammen mit der Helptaste.

* forcekey

Die Tastenkombination mit der HyperHelp_cx aufgefordert wird ein Inhaltsverzeichnis aller ihm bekannten Hilfstext im AmigaGuide-Format auf den Bildschirm zu bringen.

* sbuff

Die Größe des Ringpuffers der Eingabestromüberwachung 'watchdog'.

Achtung! Diese Einstellungen können alle mit dem Preferences-Programm

HyperHelp vorgenommen werden aber nur falls sie eine registrierte

Version von MUI besitzen. Ansonsten sind diese

Einstellungen in den Tooltypes von HyperHelp_cx vorzunehmen.

Nachdem alle Voreinstellungen richtig gesetzt sind, müssen sie eine

Index-Datei ihrer Hilfstexte erstellen. Dies geschieht

entweder mit dem Programm MakeIndex (der unbequeme Weg) oder mit dem Preferences-Programm HyperHelp.

Die Index-Datei kann mit HyperHelp erstellt werden, auch wenn sie nur eine unregistrierte Version von MUI benutzen!!

Schritt für Schritt konfiguriert

Hier nun einige Beispiel wie Sie HyperHelp auf verschiedenen Systemen konfigurieren.

Sie besitzen eine registrierte Version von MUI:

1 - Starten sie HyperHelp

2 - Setzen sie den den Name der Index-Datei ein. Dieser Name ist von Ihnen frei wählbar. Vergessen Sie aber nicht den ganzen Pfad mit anzugeben.

3 - Falls Sie mit den Voreinstellungen von CX_PRIORITY und BUFFER nicht zufrieden sind ist jetzt der Zeitpunkt gekommen diese zu ändern.

4 - Stellen Sie inder Popup-Liste ihren Editor ein. Sollte er nicht in der Liste sein, so sind zwei Dinge zu tun:

a) Setzen sie ARExx-Port-Name und ARExx-Kommando (wie oben beschrieben) in die entsprechende String-Gadgets ein.

b) Schicken sie ARExx-Port-Name und ARExx.Kommando bitte an uns, so daß wir in der nächsten Version HyperHelp um ihren Editor erweitern können.

Sollten sie Port-Name und Kommando nicht wissen, so schlagen sie bitte in der Dokumentation ihres Editors nach.

5 - In dem Sie den Save-button drücken werden die Einstellungen abgespeichert.

6 - Fügen Sie mindestens einen Hilfstext in die Index-Datei ein, indem sie entweder den Mnüpunkt 'Add index...' auswählen, oder in dem Sie

das Icon einer AmigaGuide-Datei über das Feld 'Build Index | (Drop Icon here)' ziehen.

7 - Starten Sie **HyperHelp_cx**.

Sie besitzen eine registrierte Version von MUI:

1 - Starten sie **HyperHelp**

2 - Setzen sie den den Name der **Index**-Datei ein. Dieser Name ist von Ihnen frei wählbar. Vergessen Sie aber nicht den ganzen Pfad mit anzugeben. Wichtig! Diesen Namen müssen diesen Schritt jedesmal wenn sie das Programm neu starten wiederholen. Abhilfe: Sie lassen MUI bei Stefan Stuntz registrieren.

3 - Fügen Sie mindestens einen Hilfstext in die Index-Datei ein, indem sie entweder den Mnüpunkt 'Add index...' auswählen, oder in dem Sie das Icon einer AmigaGuide-Datei über das Feld 'Build Index | (Drop Icon here)' ziehen.

4 - Setzen Sie die Tooltypes, wie oben beschrieben, im HyperHelp_cx-Icon. Wichtig ist dabei daß Sie den INDEX Tooltype auf die selbe Index-Datei wie in Schritt 1 setzen. Vergessen Sie dabei nicht den ganzen Pfad mit anzugeben. Danach müssen sie noch die Tooltypes PORT und COMMAND entsprechende den Werten die Editor benötigt setzen.

5 - Starten Sie **HyperHelp_cx**.

Sie besitzen keine Version von MUI

Dann müssen Sie leider komplett auf das sehr komfortable Preferences Programm verzichten. Als Ersatz haben wir das Programm MakeIndex beigelegt.

1 - Starten Sie das Programm MakeInde von der Shell aus und übergeben sie MakeIndex den Name ihrer **Index**-Datei. (Dieser Name ist von Ihnen frei wählbar. Vergessen Sie aber nicht den ganzen Pfad mit anzugeben.) Neben dem Name der Index-Datei müssen sie auch noch mindestens den Namen einer AmigaGuide-Datei in der Kommandozeile an MakeIndex übergeben.

2 - Setzen Sie die Tooltypes, wie oben beschrieben, im HyperHelp_cx-Icon. Wichtig ist dabei daß Sie den INDEX Tooltype auf die selbe Index-Datei wie in Schritt 1 setzen. Vergessen Sie dabei nicht den ganzen Pfad mit anzugeben. Danach müssen sie noch die Tooltypes PORT und COMMAND entsprechende den Werten die Editor benötigt setzen.

3 - Starten Sie **HyperHelp_cx**.

HyperHelp und die Shell

Alle Tooltypes können auch über die Kommandozeile an HyperHelp_cx übergeben werden. Dies ist besonder nützlich, falls Sie HyperHelp_cx aus einer Skriptdatei heraus starten wollen oder falls Sie die Workbench verachten und lieber nur mit der Shell arbeiten.

1.6 Der Voreinsteller

ACHTUNG: HyperHelp benötigt mindestens 8k Stack.

HyperHelp erfüllt zwei Aufgaben:

1. Mit HyperHelp können Sie das HyperHelp_cx konfigurieren.
2. HyperHelp dient ebenfalls dazu neuen AmigaGuide-Dateien in die Index-Datei einzubauen.

Folgende Eigenschaften von HyperHelp sollten Ihnen die Arbeit erleichtern:

- * Ein Popup Filerequester um die Index-Datei auszuwählen.
- * Eine Popup Liste mit Voreinstellungen von ARexx Port und Kommando für verschiedene Editoren.
- * Eine 'AppWindow box'. D.h. Sie können die Icons von AmigaGuide-Texten über diese Box ziehen und HyperHelp bindet diese Texten automatisch in das Hilfsystem mit ein.
- * Einen ASL Filerequester. Mit Hilfe dieses Requesters können ebenfalls AmigaGuide-Dateien in das Hilfsystem integriert werden. Mit Hilfe der Shift-Tasten und der linken Maustaste können auch mehrere Dateien gleichzeitig ausgewählt werden.
- * Zu jedem Gadget kann mit der Help-Taste ein Hilfstext eingeblendet werden.

Achtung! Die Save-Funktion funktioniert nur, falls Sie eine registrierte Version von **MUI** besitzen.

Die Gadgets von HyperHelp

Index

Priority

Buffer

Stream Buffer

HotKey

Force Key

Editor

Port

Command

Save button

Quit button

App window box

1.7 Index, NEIN kein Index nur die Erklärung zum Index-Gadget

Dieses String Gadget muß den Name der **Index**-Datei enthalten.

Dieser Name ist von Ihnen frei wählbar. Vergessen Sie aber nicht den ganzen Pfad mit anzugeben. Falls diese Datei nicht existiert so wird sie von HyperHelp erzeugt.

1.8 Die Eingabeüberwachung, Watchdog

Normally, if we know the correct port name and the correct command, we can request a word from a text editor. Most shells don't have an ARexx port, however, so we need another way of finding out what the user has been typing so that we can draw a keyword from this information. This is where WatchDog comes in.

WatchDog is a small child process of HyperHelp_cx that keeps track of all the characters you type in. When you press the hotkey, HyperHelp_cx checks to see if the active window belongs to your editor. If not, then it asks WatchDog to return what the user has been typing. HyperHelp_cx then looks at that information, which consist of every key the user has pressed (for the last X keystrokes). It then attempts to intelligently interperate any editing codes that appear in the list of keypresses, and generates a keyword from this. This method has the advantage that we can also extract keywords from string gadgets and other such one-shot sources.

It is important to remember that, although HyperHelp_cx can deal with some editing of a line, it is impossible to correctly interperate all editing codes. This is because only a few codes are standard editing commands (left, right, delete, and backspace). Most other codes that often occur (shift-left, alt-left, etc.) often vary from shell to shell. In addition, HyperHelp_cx is unable to deal with history buffers. What this means is that if you type in a word in a shell and then use the history buffer command <up-arrow>, HyperHelp_cx will not be able to extract that word from the history buffer.

HyperHelp_cx has to have some way of telling what defines a "word."

The following characters define the boundries of a word:

Spaces

Non-alpha-numeric characters

Returns

Quotes

Parenthesis

You may set the amount of keystrokes WatchDog keeps track of with the SBUFF option, or the **Stream Buffer** gadget in HyperHelp. Remember that no matter how large the buffer is, HyperHelp will only look at the last word in the stream. Since it is unlikely that you will be typing in words longer than 64 letters long (including editing commands), the default of 64 is normally adequate. More than 128 for the stream buffer verges on absurd.

1.9 priority

You may set the Commodities priority of the **HyperHelp_cx** program with this gadget. Values for this gadget must range from -127 to 127. If you do not know what effect the CX_PRIORITY of a program has, you can leave this value at its default.

1.10 buffer

The value in this gadget will effect the speed of the keyword searches, and represents the buffer for disk operations. In general, the higher the value, the faster the search. The amount of memory that **HyperHelp_cx** will use is equal to about 47k + BUFFER. This number should be no less than 200, and numbers over 10000 are excessive (and will provide increasingly less speed improvement). 5000 is a good average number, and you can experiment. If you have a very slow hard drive, you might use a higher number (5000 +); a fast SCSI-II drive could have a lower number (300-2000).

1.11 Stream Buffer

The value here determines how big the buffer for the stream should be. This buffer is where the HyperHelp system will store your keystrokes so that it can tell what you've been typing into the shell, or a string gadget, or... whatever. It is unusual that this should ever be more than 64 bytes, because it is unusual that you will have a keyword longer than 64 characters, including simple editing. 32 bytes is also a safe number. The only negative effect that can result from this value being too small is that your search musters will be truncated, and you'll get keywords like: "arianism" from "disestablishmentarianism". (Even this example has at most 24 bytes).

1.12 HotKey

With this you can set which hotkey will call HyperHelp_cx link cx}. The default is right-shift - help ("rawkey rshift help"). The word "rawkey" is not required, and you may use any legal Commodities hotkey.

1.13 Force Key

With this string you define the hotkey combination which will force HyperHelp_cx to open a guide to all of the files the HyperHelp system has access to. The default here is "right-alt-help".

1.14 Editor

This string has no actual effect on the operation of HyperHelp_cx link cx}, and is here only for reference. This string may be set by choosing an editor from the Editor list (accessed by pressing the popup button to the right of the string gadget). To choose one of the preset editors, open the list and double click on one of the editor names. The Port and Command strings will be automatically updated.

1.15 Port

This is the name of the ARexx port of your favorite text editor. To find out what this portname is you must read the documentation of your text editor. The information will likely be under a section called "ARexx", "ARexx Interface", or something similar. You do not have to set this value if you use one of the pre-set editors.

Note The Port and Command arguments are usually case sensitive. If you type these in yourself, you must make sure that you type them in exactly as they appear in the documentation.

1.16 Command

HyperHelp_cx will attempt to request a word from your text editor. Normally, this word will be the word under the cursor in your editor. The string in this gadget is the command HyperHelp_cx link cx} will send to your editor to request the word. You must check the documentation to find out what this value is. The command for

Edward and CygnusED is "GETWORD" and for GoldED "EXTRACT", for example. You do not have to set this value if you use one of the pre-set editors.

Note The Port and Command arguments are usually case sensitive. If you type these in yourself, you must make sure that you type them in exactly as they appear in the documentation.

1.17 Save

Press this button to save your configuration. This button only has effect if you have a registered version of **MUI**.

1.18 Quit

Pressing this button will end the **HyperHelp** program.

1.19 Build Index

Dropping the icons of AmigaGuide documents on this box will add the guide to the **index** file.

1.20 The index file

The index file is very important to the HyperHelp system. It is produced by either the MakeIndex (CLI) or the HyperHelp prefs} (MUI) program. Both of these programs search through the supplied (by you) AmigaGuide documents for nodes (buttons), which they then add to the index file.

The **HyperHelp_cx** program accesses this file whenever you press the hotkey and uses it to find out where each node is located.

HyperHelp_cx either tries to find a match for the supplied keyword (if any), or simply shows you all files (not all nodes) you have stored in the index.

The index file is a simple text file, and has the following format:

```
@$amigaguide_file_with_full_path
```

```
@ @node_name_1
```

```
@ @node_name_2
```

```
...
```

@@node_name_last

@\$amigaguide_file2_with_full_path

@@node_...

...and so on. Blank lines and random text in this file are accepted; however, this will obviously increase the amount of time to search for keywords in the text.

In general, you shouldn't modify this file, or try to generate it yourself. Rather, let one of the supplied programs create and maintain it for you. There are two exceptions to this: if you wish to remove one guide from the index file, you can do this by hand with a text editor. Be sure to remove the file name and all nodes associated with it. The second case is when you must change the path or name of an amigaguide file. This you can also do by hand. In this case, be sure that you enter the full path name. If you need to change many path names, it may be simpler to simply regenerate the entire index. However, a "search-and-replace" from your text editor will also do the job.

If **HyperHelp_cx** has any problems with an index file that you've modified, the easiest solution is to simply regenerate the index file with MakeIndex or **HyperHelp**.

In addition to the index file is the index guide. This file (which has the same name as the index file with a "_gd" appended) contains a list of all the guides the HyperHelp system has access to, organized into directories. This greatly speeds up the access times on some searches, provides almost immediate response to the ForceKey forcekey}, and provides a more structured view of the system. It is suggested that you not have all of your guides in the same directory, but that you divide them up into sub-directories which represent the domain of the group of files therein. For example, I have all of the RKR Autodocs (converted into AmigaGuide format, of course) in a directory called "Eris:RKM-Autodocs/" and all of my MUI Autodocs in "Eris:MUI-Autodocs". You will find that if you place all of your guides in one directory that it will soon become difficult for you to find particular guides in the index.

1.21 Das Arbeitstier

Using HyperHelp_cx

HyperHelp_cx ist das Arbeitstier des HyperHelp-Systems. HyperHelp_cx wartet im Hintergrund bis sie eine der beiden Hotkey Kombinationen drücken.

Sie können HyperHelp_cx auf 3 Arten benutzen:

1 - Sie befinden sich in ihrem Editor und möchten zu einem bestimmten Wort in ihrem Text einen Hilfstext, dazu positionieren sie den Cursor auf diesem Wort und drücken sie die standard Hotkey-Kombination (rshift-help). HyperHelp_cx fordert dann über ARexx das Wort von ihrem Editor an und durchsucht die Index-Datei nach einem passenden Hilfstext. Findet es mehrere passende Texte so könne sie aus einer Übersicht den gewünschten auswählen. Findet es keinen passenden Eintrag so erhalten sie eine Übersicht aller verfügbaren Texte.

Anmerkung: HyperHelp_cx fasst den Begriff passend recht weit. D.h. findet es keinen 'richtig' passenden Eintrag so entwickelt es hin und wieder recht viel Phantasie.

2 - Falls sie sich nicht in ihrem Editor befinden und die standard HotKey-Kombination benutzen, so versucht HyperHelp_cx mittels des **WatchDogs** das letzte Wort das sie eingegeben haben zu erkennen. Dabei ist es im Gegensatz zu seiner Konkurrenz so intelligent auch die einfachen Editiertaste wie Cursor links/rechts Backspace und Delete zu erkennen. Dank des Watchdogs ist HyperHelp in der Lage aus allen Situation, wie z.B. String-Gadgets, Shells usw., ein Schlüsselwort zu erkennen.

3 - Sie wollen keine Hilfe zu einem bestimmten Begriff, sondern eine Übersicht aller eingebundenen AMigaGuide-Dateien. Dazu müssen sie nur die Tatenkombination 'Forcekey' drücken. (voreingestellt: ralt-help)
Neben diesen Möglichkeiten besitzt HyperHelp_cx noch einige zusätzliche Möglichkeiten:

a) Sie können optional einstellen, daß sie jedes Schlüsselwort in einem kleinen Requester zuerst editieren können bevor HyperHelp_cx es zur Suche nach einem Hilfstext benutzt.

b) Bei der Suche unterstützt HyperHelp_cx die Wildcards '*' und '?'.
'*' steht dabei für die Kombination beliebig vieler Zeichen.

'?' An der Stelle '?' kann der Suchalgorithmus ein beliebiges Zeichen einfügen.

1.22 Rights and reservations

Das Copyright ©1994 von 'HyperHelp' liegt bei Apocalypse Productions. Dabei bezieht sich 'HyperHelp' auf alle im Packet enthaltenen Programme und Dateien.

Das HyperHelp Archiv ist 'freely distributable'. Dies bedeutet, daß es von jedem kopiert werden darf solange er nicht mehr als eine angemessene Kopiergebühr dafür verlangt. Die Autoren behalten sich aber alle Rechte an Quellcode und Programmen vor.

Weder die Programme noch der Quelltext (oder Teile davon) dürfen ohne eine schriftliche Genehmigung der Autoren in kommerziellen Programmen benutzt werden.

Weder die Programmierer noch Apocalypse Productions haften für irgendwelche Schäden oder Folgekosten die durch die Anwendung von 'HyperHelp' entstehen. Es wird keine Garantie übernommen daß die Programm des HyperHelp-Paketes fehlerfrei sind.

--

We will not be held responsible if HyperHelp reformats your harddrive or blows up your monitor, or if the very existance of HyperHelp on your HD causes your friends to avoid you. I am not liable if the Mafia come and snuff you while HyperHelp is running, or if you use HyperHelp to help you break into National Security Council network or rob a bank. Since HyperHelp only runs under OS2.0 or higher, you can't blame me if you get an inferiority complex about having an earlier version of the OS and spend the rest of your life in a metal institution. But if any of this stuff happens, let me know so I can brag about it in the next version.

--

Da es absolut sinnlos ist von irgendjemand etwas zu fordern ohne seine Forderung auch durchsetzen zu können und Sean und ich eine Abneigung gegen 'Crippledware' haben enthält dieses Archiv die Vollversion von HyperHelp. Trotzdem werden wir uns natürlich über jede Mark freuen die wir für HyperHelp erhalten. Da HyperHelp aber schon als Vollversion ausgeliefert wird, kann ich jedem der sich registrieren läßt nur eine Postkarte und Informationen über neue Version von HyperHelp versprechen. Alle die sich entschliesen uns eine Zuwendung zu schicken sollten sie dem Programmierer schicken der ihnen am nächsten ist. (Sean für Amerika und Helmut für Europa) Natürlich sind wir auch an ihrer Meinung, ihren Wünschen, ihren Problemen, Visionen oder wenss denn sein muß ihrer Kritik interssiert.

(Am liebsten natürlich via EMail)

--

Which brings us to the plug:

This program is SHAREWARE! If you use this program, you must send me \$6000, so that I can buy a new Amiga 6000/060, when they come out. If you don't, the ETHICS POLICE will come to take you away!! (That's a joke, Bob.)

REAL programmers don't charge shareware fees.

REAL programmers program for the sheer joy of programming, not from some inbred capitalistic instinct.

Whenever you write a program of worth and give it away for FREE (you may, of course, retain your rights to the code), you get Karma points, which you can then use to buy your way into Programmer Heaven, where laptop Crays are free, and the television frequency wavelengths are occupied by a no-charge ICDN network.

IF you charge shareware fees, you get sent to The Bad Place, where you are forced to work round the clock on used Sinclair PCs with pressure keypads, programming accounting packages and missile guidance systems.

--

1.23 The CLI index creator

In the case that you don't have MUI, or are simply a fanatical CLI user, a program for generating **index** files is included in the HyperHelp package. This program is called MakeIndex, and takes a number of arguments which it uses to generate an index file.

If you run MakeIndex with no arguments you will get the following:

USAGE: makeindex <index> {files}

If you supply a question mark you will receive the standard template:

I=INDEX/A,F=FILES/M:

INDEX - This parameter is required. The INDEX is a complete path and file name of the index file you are using. If this file does not exist, it will be created. If the file does exist, the additions will be appended. Only one index file can be used by the commodities program at a time.

FILES - You may supply the names of the guides you wish to add to the index. If you do not supply this parameter, MakeIndex will open an ASL filerequester for you to choose your amigaguides with.

1.24 The CLI preferences program

You'll probably only be using this program if you find yourself using MakeIndex as well. This program sets the preferences file which simplifies using HyperHelp_cx. You do not need to use this program if you have a registered copy of MUI. The program HyperHelp provides a much more intuitive interface to do this job. This program will generate a preferences file in ENV: based on the information you give it. Note that if you use this program several things will happen:

- 1 - Any prefs file you previously had will be completely written over. Not only that, but it will be deleted first, denounced as a heratic, called bad names and then be written over.
- 2 - HyperHelp_cx will use those preferences in favor of CLI and icon arguments.
- 3 - Using this program will set all of the preferences, not just the ones you specify. Arguments that you do not specify will be written with the defaults.

The only argument you are not allowed to not enter is the **index**. This must be the full path and file name.

All of the other arguments have defaults.

1.25 Die Programmierer

Sean E. Russell -

Nach einem längeren Aufenthalt hier in Deutschland, von dem ich (Lutz) leider nur die letzten eineinhalb Monate wusste, wird Sean wieder in die Staaten zurückgehen um sein letztes Jahr an der Universität zu beenden. Was er an der Uni studiert läßt sich leicht erraten: Computer Science. Leider ist er bedingt durch den Wohnungswechsel in nächste Zeit nur schwer erreichbar.

Die einzige sichere Adresse:

Sean Russell

2906 NW Audene Drive

Corvallis, OR 97330

USA

Man sollte sich aber darauf einstellen, daß die Antworten ein bisschen länger dauern.

Helmut Lutzenberger -

Auch ich bin gerade dabei umzuziehen, aber nicht in die Staaten auch nicht nach Bornholm (leider), sondern nur innerhalb von München in eine neue (alte) Wohnung.

Am einfachsten und schnellsten bin ich über Internet zu erreichen:

lutzenbe@informatik.tu-muenchen.de

Alle Mails bezüglich HyperHelp werde ich natürlich an Sean forwarden.

Jeder der keinen Netzzugang hat kann mich über eine der folgenden Adressen erreichen:

Helmut Lutzenberger

Bunsenstr. 24

81735 München

Germany

Oder einfach im Flex oder in einem der Irish Pubs in München.

1.26 Bugs

Das HyperHelp System wurde gleichzeitig auf den folgenden System entwickelt und getestet:

* Amiga 4000/EC030

4MB RAM

80MB AT/IDE Festplatte

Maxon C++ 1.10.3

GoldED 0.98

OS 3.0

* Amiga 2000

68030 Turbokarte

9MB ram

200MB ALF3/SCSII Festplatte

Picasso II Grafikarte

SAS/C Version 6.51

CygnusED Version 2.11

OS 2.1

HyperHelp wurde zusätzlich getestet auf einem:

* Amiga 500

5 MB RAM

50MB Oktagon500/SCSII Festplatte

OS 2.1

Bug-Liste und Verbesserungen

(b) ein Bug

- (i) eine geplante Verbesserung
- (u) Nebenwirkungen und Verbesserungen welche wir dringend umsetzen sollten
- (b) HyperHelp verliert bei jedem Fenster das geöffnet wird 184 Bytes. Dies ist kein Fehler von HyperHelp! Dies ist in Fehler von MUI. Stefan Stuntz ist sich dieses Fehlers in MUI bewußt und arbeitet daran.
- (i) Mehrere Teile von HyperHelp_cx sollten in Assembler geschrieben werden. Vor allem der Levenstein-Such-Alogrithmus. (Bis jetzt ist nur der 'WatchDog in Assembler')
- (i) HyperHelp_cx sollte erkennen, falls HyperHelp die Voreinstellungen ändern und diese übernehmen. Bis jetzt übernimmt HyperHelp_cx die Voreinstellungen erst bei einem Neustart.
- (i) Ein von MUI unabhängiges Preferences Programm mit grapischer Benutzeroberfläche, damit auch alle die kein MUI haben in den Genuß eines bequemen Voreinstellers kommen.
- (i) Ein 'entferne' Gadget in HyperHelp um HyperHelp_cx von HyperHelp aus zu beenden.
- (i) Ein eigener ARexx-Port von HyperHelp_cx um das Commodity von anderen Programmen aus 'fernsteuern' zu können.

1.27 About Magic User Interface

Diese Applikation verwendet das
MUI - MagicUserInterface

(c) Copyright 1993 by Stefan Stuntz

MUI ist ein System zum Generieren und Verwalten einer graphischen Benutzerschnittstelle. Mit Hilfe eines Voreinstellerprogramms besitzt der Benutzer die Möglichkeit, das Aussehen nach seinem persönlichen Geschmack einzustellen.

MUI wird als SHAREWARE vertrieben. Um das komplette Paket zu erhalten, das sowohl eine große Anzahl an Beispielen, als auch weitere Informationen bezüglich Registrierung enthält, halten sie in ihren lokalen Bulletin Boards oder auf Public-Domain-Disketten nach einer Datei mit Namen "muiXX.lha" (XX steht für die Version-nummer) ausschau.

Wenn sie sich sofort registrieren lassen wollen, senden sie einfach
DM 20.- oder US\$ 15.-

an

Stefan Stuntz

Eduard-Spranger-Straße 7

80935 München

Deutschland

1.28 Dankeschön

An dieser Stelle an dickes 'Dankschön' an:

Asher Royce - For proofreading and distraction. Asher is a tremendous help to developers because he's one of those people who cause computers to crash by simply being in the same room as them. Since programmers unconsciously avoid bugs in their own programs, people like Asher are invaluable for the debugging process.

Karin Niggel - For being there, and rather extreme methods of distraction.

Gerhild Farcher - ...sie hat mir mit viel Geduld erklärt was in meinem neuen Garten Unkraut ist und was man Essen kann. Ich bin nur gespannt was ich ihm Herbst ernten werden.

Manfred Platz - ...dafür, daß er mir stundenlang LA und WiWi erklärt hat und dann noch für ein paar Bierchen Zeit hatte.

Prof. Kroll - ...für seine endlose Geduld.

Ulla Koppenhagen und Volker Heun für die vielen vielen Liter Kaffee und CylonTee.

Guinness - ...für ein geniales Bier.

e tutti gli altri che hanno aiutato a realizzare questo progetto

1.29 You did WHAT?!?!?

It's recently (04-May-94) been announced in Germany that Commodore USA has been bought out by some random consortium.

Normally we would say that this is Not Good. However, as long as the OS stays the same (with, er, minor improvements such as RTG and memory protection allowed), this could be a move for the better. Certainly there exists no company in the world with a worse marketing strategy than Commodore. Except for Atari, but it was run by people who got kicked out of Commodore, so what do you expect?

In any case, assume for a moment the worst: This is the End.

No more support. (Did we ever really have any???)

This is the best time to really start supporting those shareware programmers. Already, the majority of good software out there is

share/gift/freeware or PD. Due to the recent Bad News, this situation may become even more extreme. Since the people who are producing (some of) the best software for your computer, it would be wise if you supported them. Especially us Americans.

Did you know that Americans are the single worst source of shareware contributors, world-wide? We have over three times the amount of people in the States than are in Germany (250 million vs. 80 million), and many shareware producers can count 10 times as many contributors from Germany than the States. We're well known as the cheapest users in the world.

I'm not making a plug for you to send money to us. It'd be nice, but we program for the love of programming, and as long as Amiga OS exists, we'll be making software for it. But there are many damned good programmers out there producing damned good software who need encouragement.

Invest in shareware... it's the future.
