

Mehr Power mit Winhelp-Makros

Durch den Einsatz der in Winhelp integrierten Makro-Sprache läßt sich Windows-Hilfe mit eigenen Funktionen erweitern und maßschneidern. Beispielsweise kann man eigene Buttons, Menüs und sogar Sounds (.WAV) und Video-Clips (.AVI) einbauen. Einen groben Überblick gibt Ihnen die Übersicht.

Winhelp-Makros lassen sich bei Öffnen der Hilfedatei, bei Anzeige einer bestimmten Hilfeseite, bei Auswahl eines Hotspots, Buttons oder Menüpunkts sowie durch einen externen Winhelp-Funktionsaufruf ausführen.

Um Winhelp-Makros zu verstehen und optimal einsetzen zu können, empfiehlt sich die in deutscher Sprache erhältliche Buch-Toolbox:

Winhelp-Makros - Befehlsreferenz, Tips und Tools

Info

Diese Hilfedatei wurde mit dem Windows Help Magician entwickelt. Sie demonstriert die in Microsoft Windows Help (Winhelp.Exe) enthaltene Makro-Sprache. Alle enthaltenen Befehle werden im Buch Winhelp-Makros - Befehlsreferenz, Tips und Tools ausführlich behandelt.

Die Hilfedatei darf in unveränderter Form verbreitet werden. Copyright ©1993 by

Zoschke Data GmbH
Bahnhofstraße 3
D-24217 Schönberg/Holstein

Tel. 04344/6166 Fax: 04344/6162 CompuServe E-Mail: 71340,2051

Außer Werkzeugen zur Hilfe-Erstellung für Windows und DOS bekommen Sie bei Zoschke Data **Tools zu Microsoft Visual Basic** für (fast) jeden Zweck. Ausführliche Informationen senden wir Ihnen auf Anfrage gerne zu (Stichwort WM-1).

Irrtum und Änderungen vorbehalten. Microsoft und MS-DOS sind eingetragene Warenzeichen, Microsoft Visual Basic und Windows Warenzeichen der Microsoft Corporation. The Windows Help Magician ist Warenzeichen von Software Interphase, Inc. Weitere Bezeichnungen sind zum Teil Warenzeichen oder eingetragene Warenzeichen.

Winhelp-Makros

Um Ihnen einen Eindruck von der Leistungsfähigkeit der Winhelp-Makros zu vermitteln, finden Sie in dieser Online-Hilfe zu allen verfügbaren Makros ein kleines Beispiel. Die Beispiele wurden nach Makro-Befehlen alphabetisch gruppiert. Bitte wählen Sie A - U



Makros A

- About - Dialogbox "Info ..." (About...) anzeigen
- AddAccelerator - Abkürzungstasten für Makros einrichten
- Annotate - Anzeige der Dialogbox "Anmerken ..."
- AppendItem - Menüpunkt im Winhelp-Menü anhängen

Makros B

- Back - In der "Bisher"-Liste um ein Thema zurückgehen
- BookmarkDefine - Aufruf des Dialogs zur Lesezeichen-Definition
- BookmarkMore - Aufruf des Dialogs zur Lesezeichen-Definition
- BrowseButtons - Menüleiste mit Browse-Buttons (<< und >>) versehen

Makros C

- ChangeButtonBinding - Funktion eines eigenen oder Standard-Buttons ändern
- ChangeltemBinding - Funktion eines eigenen oder Standard-Menüpunktes ändern
- CheckItem - Menüpunkt "abhaken"
- CloseWindow - Haupt- oder Zweitfenster schließen
- Contents - Inhaltsverzeichnis-Seite der Hilfedatei anzeigen
- CopyDialog - Dialog "Kopieren" aufrufen
- CopyTopic - Aktuelles Hilfethema in die Zwischenablage kopieren

CreateButton - Button-Leiste um eigene Buttons ergänzen

Makros D

DeleteItem - Menüpunkt im Winhelp-Menü löschen

DeleteMark - Mit SaveMark gesetzte Hilfeseiten-Marke löschen

DestroyButton - Button aus der Button-Leiste löschen

DisableButton - Button sperren und grau darstellen

DisableItem - Menüpunkt im Winhelp-Menü sperren

Makros E

EnableButton - Gesperrten Button freigeben

EnableItem - Gesperrten Menüpunkt im Winhelp-Menü freigeben

ExecProgram - Programm starten

Exit - Windows-Hilfe verlassen

Makros F

FileOpen - Dialogbox "Öffnen ..." (Open...) anzeigen

FocusWindow - Fokus auf Haupt- oder Zweitfenster setzen

Makros G

GoToMark - Zu einer mit SaveMark gesetzten Marke springen

Makros H

HelpOn - Thema "Hilfe benutzen" aus WINHELP.HLP anzeigen

HelpOnTop - Hilfefenster auf "Immer im Vordergrund" einstellen

History - "Bisher"-Liste (History) anzeigen

Makros I

IfThen - Makro bei Vorhandensein einer bestimmten Marke ausführen

IfThenElse - Je nach Vorhandensein einer Marke einen von zwei Makros ausführen

InsertItem - Menüpunkt in einem Winhelp-Menü einfügen

InsertMenu - Menü in der Winhelp-Menüleiste hinzufügen

IsMark - Vorhandensein einer bestimmten Marke prüfen

Makros J

JumpContents - Inhaltsverzeichnis der angegebenen Hilfedatei anzeigen

JumpContext - Hilfeseite mit der angegebenen Kontextnummer anzeigen (Beispiel: 2)

JumpHelpOn - Hilfeseite "Hilfe benutzen ..." (About Help ...) anzeigen

JumpId - Hilfeseite mit dem angegebenen Kontext-String anzeigen (Beispiel: "Tst1")

JumpKeyword - Hilfeseite zum angegebenen Stichwort anzeigen (Beispiel: "Info")

Makros N

Next - In einer Browse-Folge vorwärtsblättern (wie Button ">>")

Not - Ergebnis von IsMark (Prüfung auf Vorhandensein einer Marke) invertieren.

Makros P

PopupContext - Hilfeseite mit der angegebenen Kontextnr. als Popup anzeigen (Beispiel: 2)

PopupId - Hilfeseite mit dem angegebenen Kontext-String als Popup anzeigen (Beispiel: "Tst1")

PositionWindow - Hilfefenster-Größe und -Position einstellen

Prev - In einer Browse-Folge rückwärtsblättern (wie Button "<<")

Print - Aktuelle Hilfeseite drucken (auch Zweitfenster)

PrinterSetup - Windows-Dialog "Druckereinrichtung..." aufrufen

Makros R

RegisterRoutine - DLL-Routine zum Aufruf in Hilfe-Makros registrieren

RemoveAccelerator - Abkürzungstasten-Zuweisung entfernen

Makros S

SaveMark - Hilfeseiten-Marke setzen

Search - Dialogbox "Suchen" (Search) anzeigen

SetContents - Neue Inhaltsverzeichnis-Seite einstellen

SetHelpOnFile - Andere Hilfedatei für "Hilfe benutzen" angeben

Makros U

UncheckItem - Menüpunkt-Häkchen entfernen

Undokumentierte Goodies

AddAccelerator

RemoveAccelerator

Mit AddAccelerator lassen sich Abkürzungstasten bzw. Tasten-Kombinationen definieren, bei deren Betätigung ein Makro ausgeführt wird. Wählen Sie Ausprobieren, um die Tastenkombination **Ctrl-Alt-F5** für den Copy-Dialog vorzusehen. Drücken Sie dann diese Tastenkombination, um die Zuweisung zu testen!

Mit **RemoveAccelerator** läßt sich die Abkürzungstasten-Definition wieder löschen. Wählen Sie Ausprobieren und drücken Sie nochmals **Ctrl-Alt-F5**! Die Tastenkombination ist jetzt wieder inaktiv.

Menüs hinzufügen

Mit dem Makro **InsertMenu** können Sie die WinHelp-Menüleiste um eigene Menüs erweitern. Mit dem Makro **AppendItem** fügen Sie dem Menü Einträge hinzu, durch deren Auswahl wiederum ein anderer Makro ausgeführt werden kann. Wählen Sie Ausprobieren, um ein Beispiel-Menü namens **Ergänzungen** mit zwei Menüpunkten zu erzeugen!

Beachten Sie, daß Sie "Ausprobieren" hier nur einmal aufrufen können. Bei wiederholtem Aufruf stellt WinHelp fest, daß ein Menü dieses Namens schon vorhanden ist und gibt eine entsprechende Meldung aus. Im "richtigen Leben" läßt man Menüs daher nicht in Sprüngen, sondern meist gleich beim Öffnen der Hilfedatei einrichten (Makros können nämlich auch beim Öffnen der Hilfe sowie beim Anzeigen eines bestimmten Hilfethemas ausgeführt werden).

AppendItem

Mit diesem Makro können Sie an ein Winhelp-Standardmenü oder ein mit InsertMenu selbst hinzugefügtes Menü einen neuen Menüpunkt anhängen, bei dessen Auswahl sich wiederum Makros ausführen lassen. Wählen Sie Ausprobieren, um am Ende des Menüs **Hilfe** einen Menüpunkt anzuhängen!

Beachten Sie, daß Sie "Ausprobieren" hier nur einmal aufrufen können. Bei wiederholtem Aufruf stellt WinHelp fest, daß ein Menüpunkt dieses Namens schon vorhanden ist und gibt eine entsprechende Meldung aus. Im "richtigen Leben" läßt man Menüs daher nicht in Sprüngen, sondern meist gleich beim Öffnen der Hilfedatei einrichten (Makros können nämlich auch beim Öffnen der Hilfe sowie beim Anzeigen eines bestimmten Hilfethemas ausgeführt werden).

InsertItem

Mit diesem Makro können Sie in ein Winhelp-Standardmenü oder ein mit InsertMenu selbst hinzugefügtes Menü einen neuen Menüpunkt einsetzen, bei dessen Auswahl sich wiederum Makros ausführen lassen. Wählen Sie Ausprobieren, um das Menü **Datei** um einen Menüpunkt zu erweitern!

Beachten Sie, daß Sie "Ausprobieren" hier nur einmal aufrufen können. Bei wiederholtem Aufruf stellt WinHelp fest, daß ein Menüpunkt dieses Namens schon vorhanden ist und gibt eine entsprechende Meldung aus. Im "richtigen Leben" läßt man Menüs daher nicht in Sprüngen, sondern meist gleich beim Öffnen der Hilfedatei einrichten (Makros können nämlich auch beim Öffnen der Hilfe sowie beim Anzeigen eines bestimmten Hilfethemas ausgeführt werden).

Back

Mit diesem Makro gehen Sie in der "Bisher"-Liste (History) um ein Thema zurück. Dies können Sie jetzt ausprobieren.

CreateButton, DestroyButton DisableButton, Enablebutton

Mit **CreateButton** können Sie eigene Buttons erzeugen und der Menüleiste von Windows Help hinzufügen. Der bei der Definition angegebene Makro wird bei Betätigung des Buttons ausgeführt. Wählen Sie Button hinzufügen, um die Menüleiste mit einem "Notizen..." betitelten Button zum Aufruf des Notizblock-Programms (Notepad.Exe) zu erweitern!

Testen Sie nun das Sperren des Buttons mit **DisableButton**, indem Sie Sperren wählen. Anschließend können Sie das Freigeben des Buttons mit **EnableButton** testen, indem Sie Freigeben wählen.

Mit **DestroyButton** können Sie einen selbstdefinierten Button wieder entfernen. Um auch diesen zu testen, wählen Sie NACH dem Anlegen des Buttons und dessen Ausprobieren Button entfernen.

ChangeButtonBinding

Hiermit läßt einem Button der Winhelp-Menüleiste eine andere Funktion zuweisen. Dies funktioniert sowohl mit Winhelp-Standard-Buttons als auch mit selbstdefinierten Buttons! Wählen Sie jetzt Ausprobieren, ändert dies den Winhelp-Standard-Button "Inhalt" (Contents) so, daß er bei Betätigung die Hilfeseite mit unserem Copyright anzeigt!

Kehren Sie dann mit dem Button "Zurück" (Back) zu dieser Seite zurück und wählen Sie alte Funktion restaurieren, um mit dem Button wieder das Inhaltsverzeichnis anzeigen zu können (hiermit weisen wir dem Button den Makro "Contents" zu).

SetHelpOnFile

Mit diesem Makro lässt sich eine andere Hilfedatei als WINHELP.HLP angeben, die bei Auswahl des Menüpunkts "Hilfe benutzen" (How to Use Help) angezeigt wird. Wählen Sie Ausprobieren, um WINHELP.HLP durch WINFILE.HLP (Dateimanager-Hilfe) zu ersetzen! Testen Sie die Änderung dann, indem Sie Menüpunkt "Hilfe benutzen" wählen.

DisableItem, EnableItem, DeleteItem

DisableItem sperrt einen Menüpunkt, der der Menüleiste mit dem Makro AppendItem oder InsertItem hinzugefügt wurde; optisch wird dies durch eine graue Darstellung der sonst schwarzen Beschriftung verdeutlicht. Wählen Sie Sperren, um im Hilfe-Menü den Menüpunkt "Hilfe benutzen" (How to Use Help) zu sperren!

EnableItem gibt einen gesperrten Menüpunkt wieder frei. Sofern Sie testhalber den Menüpunkt "Immer im Vordergrund" gesperrt haben, können Sie ihn jetzt wieder freigeben, indem Sie freigeben wählen!

Mit dem **DeleteItem** können Sie in einem Winhelp-Standardmenü oder in einem mit InsertMenu selbst hinzugefügten Menü einen Menüpunkt entfernen. Wählen Sie löschen, um den Menüpunkt "Immer im Vordergrund" (Always on Top) zu entfernen!

Anmerkung: Das Entfernen gilt nur für die Dauer der Benutzung dieser Hilfe - in anderen Hilfen steht dieser Menüpunkt selbstverständlich wieder zur Verfügung. Und natürlich besteht auch die Möglichkeit, entfernte Einträge wiederherzustellen.

CheckItem

Mit CheckItem lässt sich ein dem Winhelp-Menü mittels AppendItem oder InsertItem hinzugefügter Menüpunkt "abhaken"; dies wird durch ein Häkchen-Zeichen (Checkmark) vor dem Menüeintrag angezeigt. Mit UncheckItem lässt sich das Häkchen wieder entfernen (dieser Schritt ist "rein optisch"; tatsächliche Funktionsänderungen werden separat vorgenommen).

Da das Abhaken nicht nur bei eigenen, sondern auch bei Winhelp's Standard-Menübefehlen funktioniert, können Sie dies mit "Hilfe benutzen ..." (How to Use Help) ausprobieren:

CheckItem für "Hilfe benutzen ..."

UncheckItem für "Hilfe benutzen ..."

Sehen Sie sich zwischendurch den Menüpunkt "Hilfe benutzen ..." an!

ChangeltemBinding

Hiermit läßt sich die Funktion eines Menüpunkts, die einem Winhelp-Menü mit dem Makro AppendItem zugewiesen wurde, durch eine andere ersetzen. Dies funktioniert auch mit dem Standard-Menüpunkt "Hilfe benutzen" (How To Use Help). Wählen Sie Ausprobieren, um den Winhelp-Standard-Menüpunkt "Hilfe benutzen" (How To Use Help) so zu ändern, daß er bei Betätigung die Hilfeseite mit unserem Copyright anzeigt.

Nachdem Sie den geänderten Menüpunkt "Hilfe benutzen" getestet haben, sollten Sie durch Drücken des Buttons "Zurück" (Back) hierher zurückkehren. Wählen Sie dann rückgängig machen, um dem Menüpunkt seine ursprüngliche Funktion zurückzugeben.

CloseWindow

Mit diesem Makro schließen Sie das im Aufruf angegebene Fenster (Haupt- oder Zweitfenster).
Wählen Sie Ausprobieren, um dieses Zweitfenster zu schließen!

CopyTopic

Dieser Makro kopiert das angezeigte Hilfethema in die Windows-Zwischenablage. Wählen Sie zunächst Ausprobieren und schauen Sie dann in Ihre Zwischenablage!

Test1

Dies ist Testseite **1** ...

Sie hat den Kontext-String "Tst1" und die Kontextnummer 3.

Test2

Dies ist Testseite 2 ...

SaveMark
GotoMark
DeleteMark

Neben der Verwendung von Lesezeichen (Bookmarks) bietet die Winhelp-Makrosprache eine weitere Möglichkeit, Hilfeseiten mit "Marken" zu versehen, die einen beschreibenden Text enthalten können. Im Gegensatz zu Lesezeichen werden die Marken bei Verlassen von WinHelp nicht gespeichert; dies hilft "Datenmüll" vermeiden, wenn ein Speichern nicht erforderlich ist. Zum Setzen, Anspringen und Löschen solcher Marken stehen die oben genannten Makro-Befehle zur Verfügung. Mit weiteren Makros läßt sich das Vorhandensein einer Marke prüfen und für die bedingte Ausführung anderer Makros nutzen.

Wählen Sie Marke setzen, um den Marker "Test1" zu setzen (**SaveMark**). Außerdem haben wir beispielhalber die Tastenkombination **Ctrl-Alt-F9** mit dem Makro **GotoMark("Test1")** belegt, damit Sie von jedem anderen Hilfethema aus zu dieser Seite springen können. Gehen Sie zum Beispiel zur Inhaltsverzeichnis-Seite und und probieren Sie es aus!

Wählen Sie Marke löschen, um die Marke "Test1" zu löschen (**DeleteMark**). Gehen Sie zum Beispiel zur Inhaltsverzeichnis-Seite und überzeugen Sie sich, daß sie mit Ctrl-Alt-F9 nicht mehr angesprungen werden kann.

ExecProgram

Mit diesem Makro ist es möglich, aus der Hilfedatei heraus eine Anwendung zu starten. Hier zwei Beispiele zum Ausprobieren:

Notizblock (notepad.exe)

Uhrzeit (clock.exe)

Taskmanager (taskman.exe)

Die genannten Programme müssen natürlich auf dem System vorhanden sein.

Selbstverständlich können Sie für den Aufruf von Programmen auch Buttons oder Menüpunkte vorsehen, um sie von jeder Hilfeseite aus aufrufen zu können.

Beim Makro-Aufruf lässt sich angeben, ob das Programm beim Start normal, verkleinert (Symbolgröße) oder vergrößert (ganzer Bildschirm) laufen soll. Erfordert ein Programm zusätzliche Befehlszeilen-Parameter, können Sie diese ebenfalls angeben.

Exit

Mit dem Makro Exit können Sie Winhelp (und Ihre Online-Hilfe) verlassen, als ob im Menü "Datei" (File) der Menüpunkt "Beenden" (Exit) gewählt wurde. Wählen Sie Ausprobieren, um diese Hilfe jetzt zu beenden. Aber kommen Sie zurück! Sie haben ja noch längst nicht alles gesehen!

FocusWindow

Mit diesem Makro können Sie bestimmen, welches Hilfefenster aktiv ist, indem Sie den Fokus auf das Haupt- oder Zweitfenster setzen. Zum Ausprobieren:

Fokus Hauptfenster

Fokus Zweitfenster

HelpOn

Dieser Makro zeigt die Inhaltsverzeichnis-Seite "Hilfe benutzen" (How to Use Help) der Winhelp-eigenen Hilfedatei WINHELP.HLP an. Wählen Sie Ausprobieren, um "Hilfe benutzen anzuzeigen". Mit dem Button "Back" (Zurück) gelangen sie wieder in diese Datei und zu dieser Seite.

Mit dem Makro SetHelpOnFile können Sie statt WINHELP.HLP eine beliebige andere Hilfedatei vorsehen!

HelpOnTop

Dieser Makro setzt alle Hilfefenster auf "Immer im Vordergrund" (Always On Top). Wählen Sie Ausprobieren und klicken Sie mit der Maus in einen Bereich außerhalb dieses Hilfefensters!

IfThen

IsMark

Not

Winhelp's Makrosprache eine eigene Möglichkeit, Hilfeseiten mit "Marken" zu versehen. Mit **IfThen** läßt sich das Vorhandensein einer Marke prüfen; der in IfThen angegebene Makro wird nur dann ausgeführt, wenn die angegebeneMarke existiert. Zur eigentlichen Prüfung verwendet IfThen den Makro **IsMark**, der das Ergebnis "wahr" oder "falsch" liefert. Wählen Sie IsMark ausprobieren; sofern eine Marke namens "CRIGHT" existiert, wird ein Makro zur Verzweigung zum Hilfethema "Copyright" ausgeführt. Dies passiert nicht, solange Sie nicht durch Wahl von Marke setzen (**SaveMark**) die Marke "cright" gesetzt haben ...

Mit **Not** läßt sich das IsMark-Ergebnis invertieren - aus wahr wird falsch und umgekehrt. Wählen Sie Not(IsMark) ausprobieren; sofern eine Marke namens "CRIGHT" NICHT existiert, wird ein Makro zur Verzweigung zum Hilfethema "Copyright" ausgeführt.

IfThenElse

Winhelp's Makrosprache eine eigene Möglichkeit, Hilfeseiten mit "Marken" zu versehen. Mit **IfThenElse** lässt sich das Vorhandensein einer Marke prüfen und je nach Ergebnis einer von zwei Makros ausführen. Der als erster angegebene Makro wird nur dann ausgeführt, wenn die Marke existiert; andernfalls wird der zweite Makro ausgeführt. Wählen Sie Ausprobieren; sofern eine Marke namens "CRIGHT" existiert, wird ein Makro zur Verzweigung zum Hilfethema "Copyright" ausgeführt; andernfalls wird zum Inhaltsverzeichnis gesprungen. Zu "Copyright" wird gesprungen, wenn Sie durch Wahl von Marke setzen (**SaveMark**) die Marke "cright" gesetzt haben ...

Next

Browse-Test Seite 1

Durch Aufruf des Makros **Next** blättern Sie in einer Browse-Folge zum nächsten Hilfethema (entspricht dem Drücken des Buttons ">>").

Wählen Sie Vorwärts, um zum nächsten Thema dieser Browse-Folge zu blättern. Diese Browse-Folge umfaßt diese und eine weitere (nachfolgende) Seite.

Prev

Browse-Test Seite 2

Durch Aufruf des Makros **Prev** blättern Sie in einer Browse-Folge zum vorhergehenden Hilfethema (entspricht dem Drücken des Buttons "<<").

Wählen Sie Zurück, um zum vorhergehenden Thema dieser Browse-Folge zu blättern. Diese Browse-Folge umfaßt diese und eine weitere (vorhergehende) Seite.

PositionWindow

Mit diesem Makro lassen sich Größe und Position eines Hilfe-Haupt- oder Zweitfensters einstellen.

Wählen Sie Ausprobieren, um Größe und Position des Hauptfensters und dieses Zweitfensters neu einzustellen!

RegisterRoutine

Dies ist womöglich der mächtigste Winhelp-Makro - mit ihm können Sie eine DLL-Routine registrieren und diese dann wie einen Makro benutzen!

Wählen Sie Ausprobieren, um die in der Windows-DLL "user.exe" enthaltene Routine "MessageBeep" zu registrieren und aufzurufen!

Hier ein weiteres Beispiel: Wählen Sie Ausprobieren, um mit der API-Funktion "MessageBox" ein Mitteilungsfenster anzuzeigen. Durch Übergabe einer von Winhelp bereitgestellten internen Variablen wird demonstrationshalber der aktuelle Pfad dieser Hilfedatei angezeigt.

SetContents

Mit diesem Makro können Sie ein neues **Inhaltsverzeichnis** einstellen. Wählen Sie Neu, um testweise ein anderes Hilfethema als Inhaltsverzeichnis vorzusehen. Drücken Sie dann den Button "Inhalt" (Contents), um die Änderung zu überprüfen.

Neues Inhaltsverzeichnis

Mit dem Makro **SetContents** wurde dieses Hilfethema als neues Inhaltsverzeichnis eingestellt.

Wählen Sie Wiederherstellen, um mit demselben Makro das Original- Inhaltsverzeichnis wiederherzustellen,

Print

Mit **Print** lässt sich das aktuelle Hilfethema auf dem Drucker ausgeben. Dieser Makro ist sehr wichtig, da er die einzige Möglichkeit darstellt, den Inhalt eines Zweitfensters (wie diesem hier) auszudrucken. Wählen Sie Ausprobieren, um den Inhalt dieses Fensters zu drucken!

Winhelp-Makros - Befehlsreferenz, Tips und Tools

Das etwa 200 Seiten starke deutsche Buch macht den Hilfe-Autoren mit der Syntax und Anwendung von Winhelp-Makros vertraut und geht dabei sowohl auf die Hilfe-Entwicklung mit **Microsoft Word for Windows** als auch mit dem Windows Help Magician ein. Im thematisch gegliederten Referenzteil werden alle Makro-Befehle eingehend erläutert und anhand von Beispielen demonstriert. Hinzu kommen nützliche Tips und Tricks zum erfolgreichen Einsatz; dabei wird auch so manches undokumentierte Feature aufgezeigt. Der Aufruf von Winhelp-Makros innerhalb eigener Windows-Programme wird anhand von **Microsoft Visual Basic** demonstriert.

Die **Begleitdiskette** enthält nicht nur fertige Hilfedatei-Beispiele inklusive Quelltexten, sondern auch eine DLL zur Winhelp-Erweiterung sowie eine Reihe nützlicher Tools.

[Kapitel-Übersicht](#)

[Tool-Übersicht](#)

[Highlights](#)

Winhelp-Makros - Befehlsreferenz, Tips und Tools bekommen Sie direkt bei:

Zoschke Data GmbH
Bahnhofstraße 3
D-24217 Schönberg/Holstein

Tel. 04344/6166 Fax: 04344/6162
CompuServe E-Mail: 71340,2051

Makro-Themenbereiche

Der folgende kurze Überblick zeigt Ihnen, welche wichtigen Aufgaben sich mit Winhelp-Makros durchführen lassen:

- **Hilfefenster** positionieren, in der Größe einstellen und Fensterstatus verändern (z. B. "Immer im Vordergrund").
- **Bewegen** in der Hilfedatei (Springen und Popup anhand von Kontextnummern/Strings, Stichwörtern, Lesezeichen u.a., Vor- und Rückwärtsblättern etc.).
- Eigene **Menüs** und **Menüpunkte** hinzufügen und die Funktion vorhandener ändern.
- Eigene **Buttons** hinzufügen und die Funktion vorhandener ändern. Buttons und Menüpunkte lassen sich jederzeit sperren und freigeben.
- **Winhelp-Dialogboxen** direkt aufrufen, die der Benutzer normalerweise nur über Menüs oder Buttons aktivieren kann.
- Makros **Tastendrucke** zuweisen (z. B. zum schnellen Kopieren in die Zwischenablage).
- Interne **Marken** setzen und für die bedingte Ausführung von Makros auswerten.
- Windows-Anwendungen starten sowie in DLLs enthaltene Routinen aufrufen.
- Hilfethemen direkt in die **Zwischenablage** und per **Drucker** ausgeben.

Wann immer eine Windows-Anwendung Online-Hilfe zur Verfügung stellt, geschieht deren Anzeige und Bedienung mit dem in Windows 3.x enthaltenen Programm **WINHELP.EXE**. WinHelp ist noch weit mächtiger, als viele wissen. Hauptgrund dafür ist eine eingebaute Makro-Sprache, mit der sich Windows-Online-Hilfe mit eigenen Funktionen erweitern und maßschneidern läßt.

Buch-Kapitel

Kapitel 1 gibt Ihnen einen themenorientierten **Überblick** über die Winhelp-Makrosprache und macht Sie mit der **Syntax** sowie der **Anwendung** von Makros vertraut. Der Einsatz von Makros wird sowohl für die Hilfe-Entwicklung mit **Microsoft Word for Windows** als auch mit dem Windows Help Magician (ab Version 2.0) behandelt.

Kapitel 2 dient als **Makro-Referenz** zum Nachschlagen. Thematisch geordnet werden alle Makro-Befehle eingehend erläutert und anhand von Beispielen demonstriert. Hinzu kommen wichtige Hinweise für den praktischen Einsatz.

Kapitel 3 enthält **Tips und Tricks** zum erfolgreichen Makro-Einsatz (siehe unten).

Kapitel 4 behandelt die **Utilities** und weitere Hilfsmittel für Ihre Windows-Hilfen, die Sie auf der beiliegenden Diskette finden.

Kapitel 5 zeigt Ihnen, wie Sie Windows-Hilfe - natürlich auch unter Einsatz von Makros - in **Visual-Basic-Programme** einbinden und aufrufen. Die Programmbeispiele sind im Quelltext auf der Begleitdiskette enthalten.

Der **Anhang** liefert Informationen, die für den Umgang mit den Makros nützlich sind, z. B. alphabetische Makro-Kurzübersicht, virtuelle Tastatur-Codes sowie Help-Compiler-Fehlermeldungen für Makros mit Behebungshinweisen.

Winhelp-Tools

Zum Buch gehört eine Diskette mit nützlichen Winhelp-Tools, Erweiterungen und Beispielen:

ZDHelp

ZDHelp - die Zoschke Data Hilfe-Utility - dient zum Testen von Makros und Hilfedateien. Sie verfügen damit über ein universelles Hilfsprogramm, das Ihnen bei folgenden Aufgaben zur Hand geht:

Prüfung vorhandener Hilfedateien - Sie können eigene oder beliebige fremde Hilfe-dateien öffnen, um Verzweigungen anhand von Kontext-Nummern, Kontext-Strings oder Stichwörtern auszuprobieren. Ein spezieller Debug-Modus ermöglicht es, eine Hilfedatei per Button-Druck von der ersten bis zur letzten Seite zu durchblättern.

Ausprobieren und Anwenden von Makros - Menügesteuert können Sie sechs frei definierbare Buttons mit beliebigen Makros belegen, die Sie auf die geöffnete Hilfedatei anwenden können. Die Definitionen werden für weitere Benutzungen des Programms in einer eigenen INI-Datei gespeichert. Damit können Sie sogar ohne Help Compiler, Hilfeentwicklungs-Tool und Programmierung mit Makros experimentieren und diese für sich arbeiten lassen. Versehen Sie z. B. beliebige Hilfedateien mit einem Button zum schnellen Kopieren eines Themas in die Zwischenablage!

Öffnen häufig benötigter Hilfedateien - ZDHelp "merkt" sich die vier von Ihnen zu-letzt geöffneten Hilfedateien. Wie z. B. bei Word for Windows finden Sie diese Dateien einschließlich Pfad- und Laufwerkangaben im Menü "Datei" und können sie dort einfach auswählen. Dies ist bequemer und schneller als das Öffnen von Hilfedateien über den "Öffnen"-Dialog.

ZDHCopy

Einen "Re-Compiler", der kompilierte Hilfedateien zurück ins Quellformat übersetzt, gibt es leider nicht. Mit ZDHCopy steht Ihnen jedoch ein Hilfs-programm zur Verfügung, mit dem sich der Inhalt einer beliebigen Hilfedatei (.HLP) in eine Textdatei ausgeben läßt. Die Hilfethemen werden dabei in der physikalischen Reihenfolge ausgegeben, in der sie sich - unabhängig von Browse-Sequenzen etc. - in der Datei befinden. ZDHCopy erweist sich beispielsweise als Retter in der Not, wenn versehentlich die Hilfe-Quellen gelöscht wurden; denn auf diese Weise lassen sich zumindest die Texte zurückgewinnen.

ZDMakro

Wer mit Help Magician 2.0 oder höher arbeitet, kann Makros bequemerweise per Menü auswählen. Erstellen Sie Ihre Online-Hilfe jedoch 'zu Fuß' (z. B. mit Word for Windows), können Sie sich das Leben mit ZDMakro - dem Zoschke Data Makro-Editor - erleichtern. Dieses Hilfsprogramm erlaubt ebenfalls eine menügesteuerte Makro-Auswahl und -Bearbeitung. Der Makro-Ausdruck läßt sich dann auf Knopfdruck wahlweise in die Zwischenablage oder direkt in Ihren Hilfe-Editor einfügen. Darüberhinaus erweist sich ZDMakro als nützliche Online-Befehlsreferenz für Makros.

FINFO.DLL

Durch den Einsatz von Makros kann Winhelp auf diverse andere Dateien zugreifen, z. B. auf DLLs, andere HLP-Dateien sowie WAV- und AVI-Dateien. Um Fehlermeldungen und Fehlfunktionen zur Laufzeit zu vermeiden, läßt sich mit Hilfe einer in FINFO.DLL enthaltenen **Exist-Funktion** das Vorhandensein der benötigten Dateien überprüfen, um die Hilfe gegebenenfalls zu beenden oder um entsprechende Warnhinweise zu geben. Desweiteren können Sie mit FINFO.DLL einen **Software-Schutz** vorsehen: Ihre Hilfe (oder ein Teil davon) läßt sich nur bei Vorhandensein einer Lizenzdatei benutzen!

FINFO.DLL dürfen registrierte Benutzer zusammen mit ihren kompilierten Windows-Hilfedateien lizenzgebührenfrei verbreiten.

Highlights

Hier einige Highlights aus den Tips & Tricks:

Undokumentierte Funktionen

Mit dem richtigen Know-How lassen sich auch wenig bekannte und zum Teil undokumentierte Makros und Funktionen nutzen. Wie wär's zum Beispiel mit einem verschiebbaren Popup-Menü, das sich mit Hilfe der rechten (!) Maustaste an der aktuellen Cursor-Position anzeigen läßt?

Aufruf von DLL-Funktionen

Wem die eingebauten Makros noch nicht reichen, der kann eigene DLL- und Windows-DLL-Routinen als Makros nutzen. Das Buch zeigt, wie's geht.

Multimedia

Sind die erforderlichen Audio- bzw. Video-Treiber installiert, lassen sich auch ohne weiteres beliebige Klänge (WAV-Dateien) und Video-Clips (AVI-Dateien) wiedergeben - direkt aus der Hilfedatei!

Windows Help Magician

Mit dem **Windows Help Magician 2.0** von **Software Interphase** (deutscher Exklusiv-Vertrieb: **Zoschke Data GmbH**) liegt jetzt eine stark erweiterte Version der kompletten Entwicklungsumgebung zur Erstellung von Windows-Hilfedateien vor.

Kernstück ist ein Texteditor mit integrierten Funktionen zum Einrichten von Hyper-Sprüngen und Popup-Texten, Auswählen von Schriften, Einfügen von Bitmaps, Zuweisen von Schlüsselwörtern und Definieren von Browse-Sequenzen. Schon vor dem Kompilieren ist ein Testen der Hilfe direkt in der Entwicklungsumgebung möglich; der Microsoft Help-Compiler - er ist z. B. per Download auf CompuServe erhältlich - wird automatisch gesteuert; sodaß der Hilfe-Autor nicht einmal dessen Handhabung kennen muß.

RTF- und Textdatei- Import und Export ermöglichen die Verwertung von Handbuch-Texten für die Hilfedatei und umgekehrt (RTF-Dateien enthalten auch Schriftattribute, Formatierung usw. Auch z.B. Word for Windows kann RTF-Dateien lesen und schreiben).

Zu den Highlights der neuen Version 2.0 des "Hilfe-Zauberers" gehören die völlig menügesteuerte Verwendung von Winhelp-Makros und die blitzschnelle Einzelseiten-Kompilierung für optimale WYSIWYG-Kontrolle. Windows Help Magician 2.0 ist unabhängig von Programmiersprachen einsetzbar und wird zur Zeit (8/93) noch englischem Handbuch geliefert, ab ca. 10/93 mit deutschem Handbuch.

Ab Version 2.0 bietet der Windows Help Magician von Software Interphase umfangreiche Unterstützung für Winhelp-Makros. Da die Erläuterung der Makros und ihres Einsatzes jedoch den Rahmen der Bedienungsanleitung sprengen würde, ist **Winhelp Makros - Befehlsreferenz, Tips & Tools** die ideale Ergänzung!

Undokumentierte Goodies

Bisweilen stößt man beim Erforschen einer Software auf interessante Möglichkeiten, über die man keine oder nur spärliche Informationen findet. Einige derartige Goodies werden im Buch Winhelp-Makros - Befehlsreferenz, Tips und Tools präsentiert.

Beispielsweise kennt Winhelp einen Makro-Befehl zur Rücknahme aller vorgenommenen Menü-Veränderungen und -Ergänzungen.

Oder wie wär's mit einem verschiebbaren Popup-Menü, das sich mit der rechten (!) Maustaste an der aktuellen Cursor-Position anzeigen läßt und alle möglichen Makros starten kann?

Beim Anzeigen dieser Hilfeseite wurde bereits für Sie ein solches Menü als Beispiel eingerichtet.

Drücken Sie einmal die rechte Maustaste! Können Sie sich vorstellen, wie praktisch ein solches Menü sein kann?

Da dieses Feature auf sehr ähnliche Weise auch in dem zu Winhelp.Exe aufwärtskomptiblen Programm Viewer.Exe funktioniert, dürfte man es wohl getrost einsetzen können.

