

# Win-O-Matik

Teil 2 unserer Serie beschäftigt sich mit nützlichen Erweiterungen, die Windows schon lange gefehlt haben. Tauchen Sie tiefer ein in eine Welt, in der Windows nach Ihrer - und nur nach Ihrer - Pfeife tanzt!

von peethebee



Jetzt geht es Windows ran! Im zweiten Teil unserer Autolt-Serie wird das Betriebssystem ausgebohrt und Funktionen, die man vielleicht schon lange vermisst hat, einfach selbst nachgerüstet.

Noch einmal kurz die benötigten Programme:

Autolt selbst (<http://www.autoitscript.com/download.php>), Version 3.1.1, Freeware für Windows 95, 98, NT, 2000, XP und 2003, bei Bedarf die deutsche Übersetzung der Hilfe-Datei und SciTE als komfortablen Skripteditor.

Mit unserem ersten Skript soll eine Zwei-Fenster-Ansicht - schon lange Standard in kommerziellen Dateimenagern und FTP-Programmen (und nebenbei eines ihrer besten Verkaufsargumente) - auch für den Windows-Explorer nachgerüstet werden. Legen Sie ein neues Skript an und benennen Sie es aussagekräftig. Dann beginnen wir mit der Einrichtung so genannter "HotKeys", das sind Tastenkombinationen, die bestimmte Aktionen in unserem Skript auslösen können. Wir benötigen derer drei, eine zum Aufrufen des zweiten Explorers, eine zum Wechseln zurück in die Ein-Fenster-Ansicht und eine zum Beenden des Skriptes:

```
HotKeySet("^+2", "_Explorer2")
HotKeySet("^+3", "_quit")
HotKeySet("^+1", "_CloseExplorer2")
```

Der erste Parameter bezeichnet die Tastenkombination. Dabei gilt die

gleiche Syntax wie beim Send-Befehl. Als zweiten Wert übergeben wir den Namen der Funktion, die beim Drücken der Tasten aufgerufen werden soll.

Da das Skript von oben nach unten durchgearbeitet wird, können wir nach der Einrichtung der Hotkeys problemlos eine Meldung ausgeben:

```
TrayTip("Zwei-Fenster-Ansicht", "2
Geteilte Explorer mit Strg + Shift
+ 2, Beenden mit Strg + Shift +
```

vereinfacht die Programmierung hier stark.

Damit das Skript nie zum Ende kommt (außer wenn unsere Funktion zum Beenden aufgerufen wird), bauen wir hier eine Endlosschleife ein:

```
While 1=1
    Sleep(75)
WEnd
```

Da der Ausdruck 1=1 immer wahr ist, wird die Schleife nie verlassen. Die kurze Pause dient der

Teil 1: Vorbereitung & Installationsautomation

**TEIL 2: WINDOWS UM NEUE FUNKTIONEN ERGÄNZEN**

Teil 3: Fremde Anwendungen manipulieren

Teil 4: Ideen & Ausblick

3!", 10)

Dann deklarieren wir 2 globale Variablen:

```
Global $Explorer2, $Explorer1
```

Darin werden später die Handles (Verweise) auf die Explorer-Fenster gespeichert, sodass wir sie wieder erreichen können, um ihnen weitere Befehlen zukommen zu lassen. Das Schlüsselwort Global bedeutet im Unterschied zu Local, dass auf die Variablen von überall aus zugegriffen werden kann. Lesern, die andere Programmiersprachen kennen, wird auffallen, dass hier kein Typ explizit angegeben wird. Autolt verwendet einen so genannten "Variant"-Typ und kümmert sich selbst um die notwendigen Umwandlungen. Das

Reduzierung der Prozessorlast.

Nun geht es an die Definition der Funktionen:

```
Func _Explorer2()
```

```
Func
```

Dazwischen kommen unsere Befehle:

```
Func _Explorer2()
    $Explorer1 = WinGetHandle("") ;
    kein Titel also des aktive fenster
    Run("Explorer")
    WinWaitActive("Eigene Dateien", "",
    3)
    $Explorer2 = WinGetHandle("")
    WinMove($Explorer1, "", 0, 0,
    @DesktopWidth / 2, @DesktopHeight -
    27)
    WinMove($Explorer2, "",
    @DesktopWidth / 2, 0, @DesktopWidth
    / 2, @DesktopHeight - 27)
    TrayTip("Zweiter Explorer
```

```
gestartet", "Schließen mit Strg +  
Shift + 1!", 10)  
EndFunc
```

Dieser Teil ist ein wenig komplizierter. Wir gehen davon aus, dass der Benutzer bereits den ersten Explorer offen hat, wenn er die Tastenkombination zum Aufruf des zweiten drückt. Deshalb ergibt der Aufruf von `WinGetHandle` mit einem leeren Parameter das Handle des ersten Explorers zurück, das wir

### Nützliche Funktionen

HotKeySet  
TrayTip  
ToolTip  
MsgBox  
Global  
Local  
Dim  
While...WEnd  
For...Next  
Do...Until  
WinGetHandle  
WinMove  
CDTray  
Exit

dann in der Variablen `$Explorer1` speichern. Danach wird mit der bereits bekannten Funktion `Run` ein weiterer Explorer gestartet und auf ihn gewartet. Dann ergibt erneut der Aufruf von `WinGetHandle` mit leerem Parameter das Handle des aktiven Fensters

(also des zweiten Explorers). Dieses Handle landet in der Variablen `$Explorer2`. Nun müssen die beiden Fenster noch verschoben werden: Der erste Explorer landet an der Position 0,0 (ganz links oben) mit der halben Bildschirmbreite und der Bildschirmhöhe abzüglich eines Schätzwertes für die Höhe der Taskleiste. Der Aufruf für das andere Fenster unterscheidet sich nur im x-Wert, der nun die Mitte des Bildschirms ist. Wichtig ist, dass wir hier die gespeicherten Handles über die Variablen übergeben, damit Autolt weiß, welches Fenster nun verschoben werden soll. Abschließend folgt noch eine Meldung, damit der Nutzer weiß, was passiert ist und wie er es wieder rückgängig machen kann. Diesen Weg zurück müssen wir aber erst programmieren:

```
Func _CloseExplorer2()  
    WinClose($Explorer2)  
    WinMove($Explorer1, "", 0, 0,
```

```
@DesktopWidth, @DesktopHeight-27)  
    ToolTip("Zweiter Explorer  
geschlossen!", 25, 0)  
    Sleep(2000)  
    ToolTip("")  
EndFunc
```

Zuerst wird der zweite Explorer geschlossen. Auch hier muss das Handle übergeben werden. Sollte er nicht mehr existieren, fährt Autolt einfach mit dem nächsten Befehl fort. Dieser schiebt den ersten Explorer wieder nach links oben und vergrößert ihn auf Bildschirmgröße. Eine Meldung komplettiert auch diese Funktion.

Die letzte Funktion ist auch die kürzeste:

```
Func _quit()  
    Exit  
EndFunc
```

Hier führt der einfache Aufruf von `Exit` dazu, dass das Skript sofort beendet wird.

Das war's. Testen Sie das Werk und passen Sie es bei Bedarf an Ihre Bedürfnisse an: Entfernen oder erweitern Sie die Nachrichten, die angezeigt werden, erstellen Sie eine Verknüpfung mit dem Skript im Autostart, platzieren Sie die Explorer

## Der komplette Quelltext

```
HotKeySet("^+2", "_Explorer2") ; Strg + Shift + 2  
HotKeySet("^+3", "_quit")  
HotKeySet("^+1", "_CloseExplorer2")  
  
TrayTip("Zwei-Fenster-Ansicht", "2 Geteilte Explorer  
mit Strg + Shift + 2, Beenden mit Strg + Shift + 3!",  
10)  
  
Global $Explorer2, $Explorer1  
  
While 1  
    Sleep(75)  
WEnd  
  
Func _Explorer2()  
    $Explorer1 = WinGetHandle("")  
    Run("Explorer")  
    WinWaitActive("Eigene Dateien", "", 3)  
    $Explorer2 = WinGetHandle("")  
    WinMove($Explorer1, "", 0, 0, @DesktopWidth / 2,  
@DesktopHeight - 27)  
    WinMove($Explorer2, "", @DesktopWidth / 2, 0,  
@DesktopWidth / 2, @DesktopHeight - 27)  
    TrayTip("Zweiter Explorer gestartet", "Schließen mit  
Strg + Shift + 1!", 10)  
EndFunc  
  
Func _CloseExplorer2()  
    WinClose($Explorer2)  
    WinMove($Explorer1, "", 0, 0, @DesktopWidth / 2,  
@DesktopHeight - 27)  
    ToolTip("Zweiter Explorer geschlossen!", 25, 0)  
    Sleep(2000)  
    ToolTip("")  
EndFunc  
  
Func _quit()  
    Exit  
EndFunc
```

übereinander, ...

Sie sehen, hier bestimmen Sie, was passiert, Sie sind nicht länger darauf angewiesen, das zu verwenden, was Microsoft und andere Firmen Ihnen vorsezen. Seien Sie ihr eigener Herr!

Ein weiteres, kleines Skript vereinfacht die Arbeit mit Ihren CD-Laufwerk. Sie müssen nur noch das Programm starten und die CD einlegen, das Öffnen und Schließen geschieht vollautomatisch.

TrayTip("CD-Wechsel", "Das Laufwerk

```
E: wird ausgefahren und nach 7  
Sekunden wieder eingezogen!", 10)  
sleep(1000)  
CDTray ("E:", "open");  
sleep(7500)  
CDTray ("E:", "closed");  
TrayTip("Fertig", "CD-Laufwerk  
wieder geschlossen!", 10)  
sleep(1500)  
TrayTip("", "", 10)  
Exit
```

Schon diese einfachen Skripte können Ihren Arbeitsalltag unter Windows stark erleichtern und es gibt viele weitere derartige Skripte im Internet. Zögern Sie aber auch nicht, Ihre Ideen einfach mal zu verwirklichen! Autolt unterstützt Sie dabei vorbildlich. Schicken Sie mir Ihre Werke, ich freue mich auf die Ergebnisse. (jm)

### Quellen und Links

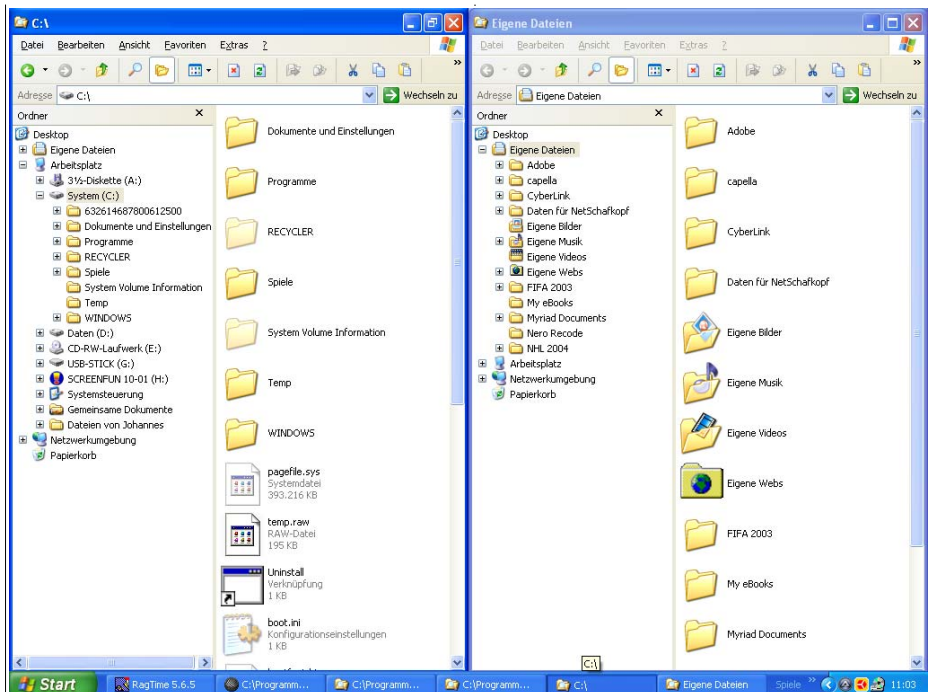
<http://www.autoitscript.com>  
<http://www.autoitscript.com/forum>

### Preview

*In der nächsten Ausgabe:*

#### **Teil 3: Fremde Anwendungen manipulieren**

*Verwirklichen Sie, was Sie sich von  
Anwendungsprogrammen wünschen:  
Funktionen, Automaten, ...*



Die nachgerüstete 2-Explorer-Funktion für den Windows Explorer im Einsatz.

## Fragen und Unterstützung

Im englischen Forum auf der Entwicklerseite unter <http://www.autoitscript.com/forum> erhalten Sie in der Regel innerhalb kurzer Zeit eine qualifizierte Antwort. Gute deutsche Foren dagegen sind leider rar gesät. Eines davon finden Sie über <http://www.autoit.de.vu>.

peethebee, der Autor dieses Artikels, steht unter der E-Mail-Adresse [peethebee@gmx.de](mailto:peethebee@gmx.de) für Rückfragen zu diesem Artikel und dem Beispielprojekt zur Verfügung. Sie können auch die ein oder andere Frage zu Ihrem Projekt an mich richten, ich kann aber natürlich weder für eine schnelle Antwort noch für eine vollständige Lösung garantieren.