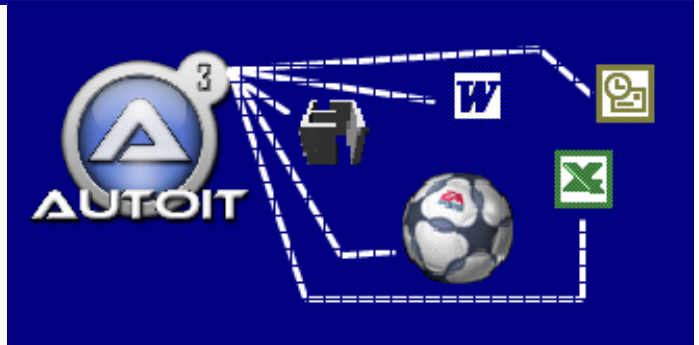


Autolt & Friends

Der dritte Teil der **Autolt**-Serie hat den Austausch mit anderen Programmen zum Thema - steuern Sie Programme fern oder tauschen Sie Daten mit Ihnen aus.

von peethebee



Wenn Sie bereits die ersten beiden Teile dieser Serie gelesen haben, wird es Sie kaum wundern, dass Autolt nicht nur in der Zusammenarbeit mit Windows glänzen kann, sondern auch andere Programme steuern und Ihnen als Anwender damit oft viel Arbeit abnehmen kann.

Noch einmal kurz die benötigten Programme:

Autolt selbst (<http://www.autoitscript.com/download>),

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.

Den Anfang soll ein kurzes Skript bilden, mit dem Sie markierten Text aus (bis auf ganz wenige Ausnahmen) jeder(!) Anwendung übernehmen und direkt an eine Internetsuchmaschine übergeben können, um den Begriff z.B. erklärt zu bekommen, oder Webseite zum Thema zu finden.

Ebenso wie das Skript aus dem zweiten Teil soll dieses in einer Endlosschleife warten, bis eine Tastenkombination gedrückt wird:

```
HotKeySet("!c", "_suche")
```

```
While 1=1  
    Sleep(100)
```

```
WEnd
```

```
Func _suche()
```

EndFunc

Der Weg führt über die Windows-Zwischenablage, über die wir den Text in unser Skript einlesen. Erst simulieren wir die Tastenkombination Strg + C, über die der Text auch von Hand in die Zwischenablage kopiert werden würde. Damit bleibt dem Nutzer sogar das Kopieren in die Zwischenablage erspart:

```
Send("^c")
```

Version unterschiedlich ist, liefert Autolt in der Konstanten @Comspec den entsprechenden Pfad. Unser Aufruf lautet also zum Beispiel für die Suche in Google:

```
Run(@Comspec & " /c start http://  
www.google.de?search=" & $text)
```

Der markierte Text wird direkt mit übergeben, sodass direkt die Ergebnissseite angezeigt wird.

Natürlich lässt sich auch dieses Skript einfach an Ihre Bedürfnisse

Teil 1: Vorbereitung & Installationsautomation

Teil 2: Windows um neue Funktionen ergänzen

TEIL 3: FREMDE ANWENDUNGEN MANIPULIEREN

Teil 4: Ideen & Ausblick

Danach holt die Autolt-Funktion ClipGet() den Text aus dem Zwischenspeicher und speichert ihn in der Variablen \$text:

```
$text = ClipGet()
```

Nun müssen wir nur noch die Webseite aufrufen. Allerdings wollen wir uns weder auf einen Browser festlegen, noch von dessen Installationsort abhängig sein. Das gute alte DOS bietet uns die Funktion start, die sich selbst darum kümmert, die angegebene Datei oder - in unserem Fall Webseite - zu starten. Dabei beachtet sie sogar die Einstellung, die den Standardbrowser festlegt. Da auch der Ort der Kommandozeile von Windows-Version zu Windows-

anpassen: Lassen Sie beispielsweise bei der englischen Google-Version suchen, suchen Sie nur auf deutschen Seiten oder verwenden Sie Ihre favorisierte Suchmaschine. Suchen Sie Übersetzungen bei dict.leo.org oder Lexikoneinträge bei Wikipedi.de.

TIPP

Unter Windows 2000/XP bietet Autolt die Möglichkeit, Fenster transparent zu gestalten. Verwenden Sie dazu die Funktion WinSetTrans() mit Werten von 0 bis 255.

Schönheitswettbewerb

Das Highlight schlechthin ist die mit Version 3 hinzugekommene Unterstützung für GUIs (Graphical User Interfaces, graphische Benutzeroberflächen). Dadurch ist es einerseits möglich, den Benutzern eines eigenen Skriptes die von Windows gewohnte Oberfläche zu bieten, deren Bedienung allgemein bekannt sein dürfte. Andererseits stehen Ihnen als Autor fast alle so genannten Controls zur Verfügung, die Sie einfach einbinden können, ohne sich um deren Darstellung oder Verhalten kümmern zu müssen. Das erledigt nämlich Windows bzw. Autolt für Sie. Im Bild auf der linken Seite sind alle verfügbaren Controls dargestellt. Als kleines Beispiel-Skript dazu soll ein kleines über den anderen Fenstern schwebendes Fenster entstehen, in das man E-Mail-Adressen eingeben (oder hineinschieben) kann, um sie auf ihre formelle Korrektheit zu überprüfen.

Bei der Arbeit mit GUIs ist es sinnvoll, die Datei GUIConstants.au3 in das Skript mit einzubinden, weil man dann im Skript mit aussagekräftigen Bezeichnungen der Optionen arbeiten kann und sich keine abstrakten Zahlen merken muss. Nach dem Prinzip der UDFs (User Defined Functions, etwa Zusatzfunktionen) ist es möglich, gesammelte Funktionen aus einer anderen au3-Datei so in das eigene Skript zu holen, als stünden sie direkt an der entsprechenden Stelle. Dadurch lässt sich viel Schreibarbeit sparen. Im Lieferumfang von Autolt sind z.B. bereits folgende UDFs enthalten (im Unterordner "Include"): Array.au3, Color.au3, Date.au3, File.au3, Inet.au3, Math.au3 und String.au3. Viele weitere UDFs sind im Forum zur Programmiersprache verfügbar, so unter anderem Prozess-, Statistik-, Arbeitsspeicher- und RTF-Funktionen oder auch Texteffekte. Solche Dateien laden Sie einfach in Ihr Include-Verzeichnis herunter und binden sie an beliebiger Stelle im Skript wie folgt ein:



Spiegeln, Spiegeln an der Wand, welches Skript ist das schönste im ganzen Land?

```
#include <GUIConstants>
```

Für unser Beispiel erstellen wir nun die Oberfläche. Dazu sind im Forum zwei interessante Programme erhältlich, die diese Arbeit etwas zu vereinfachen vermögen, deren Funktionalität aber noch nicht ausreichend ausgereift ist, um damit auch kompliziertere GUIs zu erstellen. Zuerst steht stets

```
GuiCreate(),
```

in diesem Fall gefolgt von

```
adsasd  
asd  
asd.
```

Allgemein also die Erstellung der einzelnen Controls.

Um das Fenster anzuzeigen, ist am Schluss noch ein Aufruf von

```
GUISetState()
```

nötig.

StringRegexp
als AcceptFile -> Textempfang?