

# Alles automatisch!

Erfahren Sie, wie Sie mit **Autolt** - einer einfachen, aber genialen Skriptsprache - Windows und beliebige Programme automatisieren und lange vermisste nützliche Funktionen ganz einfach selbst nachrüsten können.

## Teil 1: Vorbereitung & Installationsautomation

von *peethebee*

**N**icht immer wollen Programme so wie ihre Nutzer: Da erscheinen Bestätigungsdialoge immer und immer wieder, obwohl man den Text bereits auswendig kennt, da können Installationen von Programmen nicht unbeaufsichtigt laufen, weil sie immer wieder auf Benutzereingaben warten. Auch an Windows würde man gerne an der ein oder anderen Stelle ansetzen und sich die Wünsche, die man in seiner täglichen Arbeit mit dem System entwickelt, verwirklichen.

Die gute Nachricht: Es gibt eine einfache und geniale Skriptsprache namens Autolt, die all das möglich macht. Die schlechte: Es ist ein wenig Einarbeitungszeit notwendig, die umso kürzer ausfällt, je mehr Sie schon mit Programmierung zu tun hatten. Aber keine Angst, auch Anfänger können mit Autolt in die Programmierung einsteigen, was zusätzlich den Vorteil bietet, dass sich die ersten Erfolgserlebnisse schnell und intensiv einstellen.

### 1. Vorbereitungen

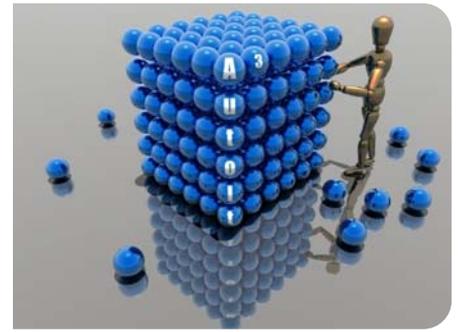
Bevor Sie mit der Entwicklung Ihrer Skripte beginnen können, brauchen Sie folgende Dateien:

- Den Skriptinterpreter an sich (<http://www.autoitscript.com/download.php>), den Sie mit der einfachen Installationsroutine in ein beliebiges Verzeichnis installieren. Dieser Artikel basiert auf der zum Readme-Schluss aktuellen Version 3.1.1. Autolt läuft unter Windows 95, 98, NT, 2000, XP und 2003. Die ein oder andere exotische

Funktion ist jedoch nicht überall verfügbar.

**Tipp:** Profis können die aktuelle Beta-Version herunterladen, die gegenüber der aktuellen Version weitere nützliche Funktionen enthält und auch stabil läuft.

- Wenn Sie des Englischen nicht so mächtig sind, laden Sie sich von der "Dokumentationsseite" am besten gleich noch die deutsche Übersetzung der Hilfe herunter.



im Explorer. Wählen Sie "Neu" und "Autolt v3 Script". Vergeben Sie einen Namen und starten Sie die Datei danach per Doppelklick. Wenn Sie SciTE4Autolt installiert haben, startet dieser Editor mit einem vorgefertigten Skript, das bereits einen Kopf enthält, in den Sie ihre Daten eingeben können. Wenn Sie das Ergebnis Ihrer Bemühungen testen möchten, klicken Sie im Explorer mit der rechten Maustaste auf Ihr Skript

### TEIL 1: VORBEREITUNG & INSTALLATIONS-AUTOMATION

Teil 2: Windows um neue Funktionen ergänzen

Teil 3: Fremde Anwendungen manipulieren

Teil 4: Ideen & Ausblick

- Prinzipiell können Sie Ihre Skripte auch mit dem bei Windows mitgelieferten Editor "Notepad" schreiben, die speziell für Autolt angepasste Version des Komforteditors SciTE ist aber speziell für größere Projekte nützlich. Es stehen auch für andere Editor-Dateien bereit, mit denen Sie Autolt-Skripte z.B. mit Syntax-Hervorhebung zu Leibe rücken können. Bei der Installation sollten Sie als Standardaktion für Dateien mit der Endung ".au3" die Bearbeitung wählen.

### 2. Erste Schritte:

Legen Sie auf Ihrer Festplatte ein Verzeichnis für Ihre Projekte an und klicken Sie darin mit der rechten Maustaste auf einen freien Bereich

und wählen den Eintrag "Run". Schneller geht es, wenn Sie im SciTE-Editor F5 drücken.

Das Highlight ist aber die Möglichkeit, aus dem Quelltext eine selbst lauffähige exe-Datei zu erstellen. Dazu wählen Sie im Kontextmenü der Datei "Compile" oder "Compile with options". In SciTE ist das entsprechende Tastenkürzel F7.

Jetzt beginnt die Arbeit an Ihrem Skript:

Im ersten Teil dieser Serie soll die Aufgabe erledigt werden, für die der englische Hauptentwickler Jonathan Bennett die Freeware ursprünglich entwickelt hat, nämlich die Automatisierung von Installationsvorgängen. Stellen Sie sich vor, Sie sollen einem Bekannten nur wegen der

Installation eines Programmes Hilfe leisten oder müssen als Administrator mehrerer Computer auf den gleichen Stand bringen.

Am besten halten Sie während der Arbeit an Ihrem Programm immer die Hilfe offen, sodass Sie stets die Parameter und ihre Bedeutung im Blick haben.

Im Folgenden betrachten wir die automatische Installation des Mozilla Firefox 1.0.2 (<http://www.mozilla.org>)

Da wir nur eine Datei mitnehmen bzw. verschicken möchten, verwenden wir die Funktion FileInstall, um die Firefox-Installationsdatei in unser Skript einzubinden.

```
FileInstall("C:\Programme\AutoIt3\
Skripte\Artikel\Firefox Setup
1.0.2.exe", "C:\")
```

Zuerst geben wir den Speicherort auf unserem Rechner an, danach den Zielort bei der Ausführung des Skriptes.

Da Firefox bei einem Update etwas allergisch auf noch laufende Firefox-Instanzen reagiert, schießen wir diese radikal ab - natürlich nicht, ohne den Benutzer vorher zu warnen und ihm damit die Chance zu geben, die Webseiten selbst zu schließen, ohne dass Daten verloren gehen:

```
MsgBox(0, "Automatische Firefox-
Installation", "Bitte schließen Sie
alle Firefox-Fenster!")
ProcessClose("firefox.exe")
```

Natürlich können Sie diese Meldung weg lassen, wenn Sie selbst die Installation durchführen und wissen, was gleich passieren wird. Danach starten wir die ausgepackte Installationsdatei:

```
Run("C:\Firefox Setup 1.0.2.exe")
```

Das führt den Befehl Run aus, der die in Klammern und Anführungsstrichen angegebene Datei startet.

Jetzt zeigt sich eine der Stärken von AutoIt: Man kann jetzt warten, bis das zu erwartende Fenster aktiv ist, unabhängig davon, wie lange es

dauert:

```
WinWaitActive("Firefox Setup")
```



### Die Information über den Abbrechen-Button im ersten Fenster der Firefox-Installation

Erst danach wird das Skript weiter abgearbeitet. Jetzt müssen wir nach und nach die einzelnen Installationsschritte abnicken. Es soll dabei verschiedene Methoden verwandt werden, um die Vielfalt der Möglichkeiten von AutoIt zu zeigen. Das Tor zu diesen Möglichkeiten öffnet das "Autolt Window Info Tool", das alle benötigten Details zu dem Fenster oder Steuerelement unter dem Mauszeiger anzeigt. Sie starten es entweder über den entsprechenden Startmenüeintrag. Da der Weiter-Button der Standard-Knopf ist, können wir zum nächsten Schritt wechseln, indem wir einfach einen Druck auf die Enter-Taste simulieren. Dazu dient der Befehl Send:

```
Send("{ENTER}")
```

Danach ist der Lizenzvertrag anzunehmen und erneut "Weiter" auszuwählen. Das können wir über die Tastenfolge "Nach-Oben-Taste", "Enter" erreichen, also:

```
WinWaitActive("Software Liz")
Send("{UP}{ENTER}")
```

Bei der Angabe des Titels des Fensters, auf dessen Erscheinen wir warten, genügt der Anfang des Titels. Dieses Verhalten lässt sich über die Funktion "Opt("WinTitleMatchMode", [Wert]) steuern. Siehe dazu die Hilfe. Im Anhang der selben finden Sie auch die Codes aller weiteren Tasten.

Man kann die Dialoge aber nicht nur mit der Tastatur steuern, sondern auch wie als "richtiger" Nutzer mit der Maus. Auch dazu bietet AutoIt umfangreiche Funktionen an. In unserem Fall führt das zu den Zeilen

```
WinWaitActive("Installationsart")
MouseClicked("left", 367, 368)
```

Der erste Parameter steht für die zu drückende Maustaste (hier die linke), die Koordinaten (x und y) liefert das Window Info Tool. Leider scheint das Fenster nicht an einer festen Position zu erscheinen, sodass wir nicht sicher sein können, den Button zu treffen. Also verschieben wir das Fenster zunächst in die linke obere Bildschirmcke und klicken erst danach:

```
WinWaitActive("Installationsart")
WinMove("Installationsart", "", 0, 0)
MouseClicked("left", 367, 368)
```

Der Parameter "Text" kann leer gelassen werden, da das Fenster schon durch den Titel eindeutig beschrieben ist.

Ein systemnäherer Weg führt über die internen Bezeichnungen der

### Nützliche Funktionen

- BlockInput
- ControlClick
- ControlSetText
- FileDelete
- FileInstall
- MouseClicked
- MsgBox
- ProcessClose
- Run
- RunWait
- Send
- Sleep
- WinClose
- WinWait
- WinWaitActive

Steuerelemente. Der Weg zu diesen Werten führt, - Sie ahnen es wahrscheinlich schon - über das Window Info Tool. Der entscheidende Wert ist die "ClassnameNN" im Abschnitt "Control Under Mouse". Sie müssen also die Maus über den Weiter-Button halten und dann die Bezeichnung ablesen.

Um diesen Wert nun zu verwenden, benutzen wir die Funktion ControlClick:

```
ControlClick("Komponenten", "", "Button7", "left")
```

Jetzt kopiert Firefox die benötigten Dateien. Danach folgt ein Bestätigungsbildschirm, auf dem standardmäßig ausgewählt ist, dass der Firefox gleich starten soll. Für eine unbeaufsichtigte Installation ist das aber unter Umständen nicht sinnvoll. Dazu sendet man einen Druck der Leertaste, da das Auswahlfeld den Fokus bereits besitzt.

```
WinWaitActive("Installation")  
Send("{SPACE}{ENTER}")
```

Zu guter Letzt löschen wir nach einer 2-sekündigen Pause noch die Installationsdatei, die wir beim Skriptstart entpackt hatten:

```
Sleep(2000)  
FileDelete("C:\Firefox Setup  
1.0.2.exe")
```

Damit ist die Installation abgeschlossen. Auf diese Weise kann nahezu jede Installation automatisiert werden. Aber auch andere häufige Vorgänge (Konfigurationen anlegen usw.) können mit AutoIt extrem vereinfacht und beschleunigt werden. Es sei auch noch auf die Programme "AU3Recorder" und "AutoItMacroGenerator" hingewiesen, die Skripte zum Teil automatisch generieren können und bei SciTE im Menü "Tools" oder separat im Forum finden. Der generierte Code kann aber leider nur in den seltensten Fällen komplett ohne Nacharbeit übernommen werden.

## Der komplette Quelltext

```
FileInstall("C:\Programme\AutoIt3\Skripte\Artikel\  
Firefox Setup 1.0.2.exe", "C:\")  
MsgBox(0, "Automatische Firefox-Installation",  
"Bitte schließen Sie alle Firefox-Fenster!")  
ProcessClose("firefox.exe")  
Run("C:\Firefox Setup 1.0.2.exe")  
WinWaitActive("Firefox Setup")  
Send("{ENTER}")  
WinWaitActive("Software Lizenz")  
Send("{UP}{ENTER}")  
WinWaitActive("Installationsart")  
WinMove("Installationsart", "", 0, 0)  
MouseDown("left", 367, 368)  
ControlClick("Komponenten", "", "Button7", "left")  
WinWaitActive("Installation abgeschlossen")  
Send("{SPACE}{ENTER}")  
Sleep(2000)  
FileDelete("C:\Firefox Setup 1.0.2.exe")
```

Die komplette Firefox-Installation dauert mit unserem Skript übrigens gerade einmal noch 17,5 Sekunden und erfordert genau einen Doppelklick auf die kompilierte Datei... (pe)

### Preview

*In der nächsten Ausgabe:*

### Teil 2: Windows um neue Funktionen ergänzen

*Schreiben Sie Skripte, die das nachrüsten, was Ihnen vielleicht schon immer in den Bereichen Komfort, Geschwindigkeit und Sicherheit an Windows gefehlt hat!*

### Quellen und Links

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

## 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.