

## Scripte

Geschrieben von: Dietmar Vollmeier

Donnerstag, den 04. Dezember 2008 um 20:38 Uhr - Aktualisiert Sonntag, den 09. Oktober 2011 um 17:10 Uhr

---

Eine Stärke von **40tude Dialog** ist, daß dieser durch Scripte *fast* beliebig erweitert werden kann. Die Scripte werden dabei in drei Kategorien unterteilt, nämlich in

- [Ereignisscripte](#)
- [Meldungsscripte](#)
- [Autonome Scripte](#)

### Ereignisscripte:

Wie der Name schon sagt, ist ein Ereignis dafür verantwortlich, damit diese Scripte ausgeführt werden. **40tude Dialog** unterscheidet dabei 13 verschiedene Ereignisse:

- [OnStartup](#)
- [OnShutdown](#)
- [OnBeforeSendingMessage](#)
- [OnBeforeSavingMessage](#)
- [OnXOver](#)
- [OnEmailChecked](#)
- [OnTimer1](#)
- [OnTimer2](#)
- [OnArticleListPaint](#)
- [OnNewsgroupListPaint](#)
- [OnWorkQueueEmpty](#)
- [OnGroupEntered](#)
- [OnOutgoingMessageChecked](#)

### Meldungsscripte:

Wann diese Scripte ausgeführt werden, läßt sich bei der Erstellung eines solchen auswählen, z.B. bei Linksklick auf die Statusleiste, usw.

Hier lassen sich 'zig verschiedene Kombinationen zusammenstellen.

## Scripte

Geschrieben von: Dietmar Vollmeier

Donnerstag, den 04. Dezember 2008 um 20:38 Uhr - Aktualisiert Sonntag, den 09. Oktober 2011 um 17:10 Uhr

---

### Autonome Scripte:

Diese können direkt aus der Scriptverwaltung, über ein Symbol in der Symbolleiste oder einfach per Hotkey aufgerufen, bzw. ausgeführt werden.

---

### Was sind INCLUDE-Files?

Damit ein Script durch den eingebunden Code nicht zu lang und damit zu unübersichtlich wird, kann und wird der Code in ein eigenes Script im Abschnitt "Eigene Scripte" ausgelagert und dann per INCLUDE in das eigentliche Script eingebunden. Das ist für dieses Script wie wenn der Code tatsächlich dort stehen würde, aber für den Programmierer ist dieses Vorgehen viel übersichtlicher.

Ich habe mir angewöhnt, die INCLUDE-Files zwecks besserer Unterscheidung und Auffindung wie folgt zu bezeichnen:

#### i\_xxx\_Name

<b>i</b>	=	damit wird ein INCLUDE-File kenntlich gemacht
<b>xxx</b>	=	Abkürzung des (Ereignis-)Scripts wo dieses INCLU
z.B. OBSavM für OnBeforeSavingMessage		
<b>Name</b>	=	Name des Script

---

### Einbindung eines INCLUDE-Files

Die Einbindung eines INCLUDE-Files in ein (Ereignis-)Script erfolgt eigentlich stets mit den selben 3 Schritten.

1. Lade dir das ZIP-File mit dem Script herunter und entpacke bei geschlossenem Dialog die darin enthaltene Datei "

## Scripte

Geschrieben von: Dietmar Vollmeier

Donnerstag, den 04. Dezember 2008 um 20:38 Uhr - Aktualisiert Sonntag, den 09. Oktober 2011 um 17:10 Uhr

---

### NameDesIncludeFiles

.ds

" in das Programmverzeichnis von

### 40tude Dialog

ins Unterverzeichnis

### ScriptsCustom

2. Starte anschließend den Dialog mit der Scriptverwaltung und binde das INCLUDE-File in das entsprechende (Ereignis-)Script ein. Gehe wie folgt dabei vor:

- Suche im (Ereignis-)Script die Zeile mit dem Code "**program xxx**;"  
xxx steht dabei für den Namen des jeweiligen (Ereignis-)Scripts, z.B. OnStartup
- Füge unterhalb dieser Codezeile das INCLUDE-File mit folgender Syntax ein  
**{ \$I NameDesIncludeFiles }**  
Program OnBeforeSendingMessage; { \$I NameDesIncludeFiles }
- Jetzt muß nur noch die Funktion eingebunden werden, damit der Code auch ausgeführt wird. Wie die Funktion heißt und mit welchen Parametern diese eingetragen werden muß, wird bei den Infos zu den jeweiligen INCLUDE-Files stets mit angegeben.
- Suche die Zeile mit dem Code "**End**";", den mit dem **Strichpunkt/Semikolon** am Ende
- Füge oberhalb dieser Zeile die Funktion ein, die ausgeführt werden soll  
Begin //Your code goes here //Return false to prevent sending the passed message result := true;  
FunktionDesIncludeFiles (Message, IsEmail); End; Begin End.

## Was tun, wenn an den angegebenen Positionen schon etwas steht?

In den meisten Fällen findest du an den angegebenen Positionen schon ein INCLUDE oder einen Code vor. Füge dann die neuen Codezeilen direkt über oder unter die vorhandenen Zeile(n) ein.

```
Program OnBeforeSendingMessage; { $I NameDesIncludeFiles1 } { $I
NameDesIncludeFiles2 } Function OnBeforeSendingMessage( Var Message : TStringlist;
Const Servername : String; Const IsEmail : Boolean ) : Boolean; Begin //Your code
goes here //Return false to prevent sending the passed message result := true;
FunktionDesIncludeFiles2(Message, IsEmail); FunktionDesIncludeFiles1(Message); End; //
function OnBeforeSendingMessage() Begin End.
```

## Scripte

Geschrieben von: Dietmar Vollmeier

Donnerstag, den 04. Dezember 2008 um 20:38 Uhr - Aktualisiert Sonntag, den 09. Oktober 2011 um 17:10 Uhr

---

Das wars dann eigentlich auch schon. Jetzt noch das (Ereignis-)Script **kompillieren** und gegebenenfalls

**ausführen**

- fertig.

---

## Beim Download war im ZIP-File eine INI-Datei enthalten, wohin gehört diese?

In einer INI-Datei stehen Informationen die das dazugehörige Script benötigt bzw. Daten die dieses Script speichert.

All diese INI-Dateien gehören standardmäßig ins Programmverzeichnis von **40tude Dialog** kopiert bzw. werden vom Script dort angelegt.

---

## Handhabung von autonomen Scripten

Diese erfolgt ähnlich einem INCLUDE-File

1. Lade dir das ZIP-File mit dem Script herunter und entpacke bei geschlossenem Dialog die darin enthaltene Datei "NameDesScriptes

.ds

" in das Programmverzeichnis von

**40tude Dialog**

ins Unterverzeichnis

**ScriptsCustom**

.

2. Starte anschließend den Dialog und die Scriptverwaltung. Suche darin das Script und kompilliere es.

3. Dem Script können jetzt noch Symbol und Beschreibung zugewiesen werden

---

## Scripte

Geschrieben von: Dietmar Vollmeier

Donnerstag, den 04. Dezember 2008 um 20:38 Uhr - Aktualisiert Sonntag, den 09. Oktober 2011 um 17:10 Uhr

### Einstellungen für aktives Skript

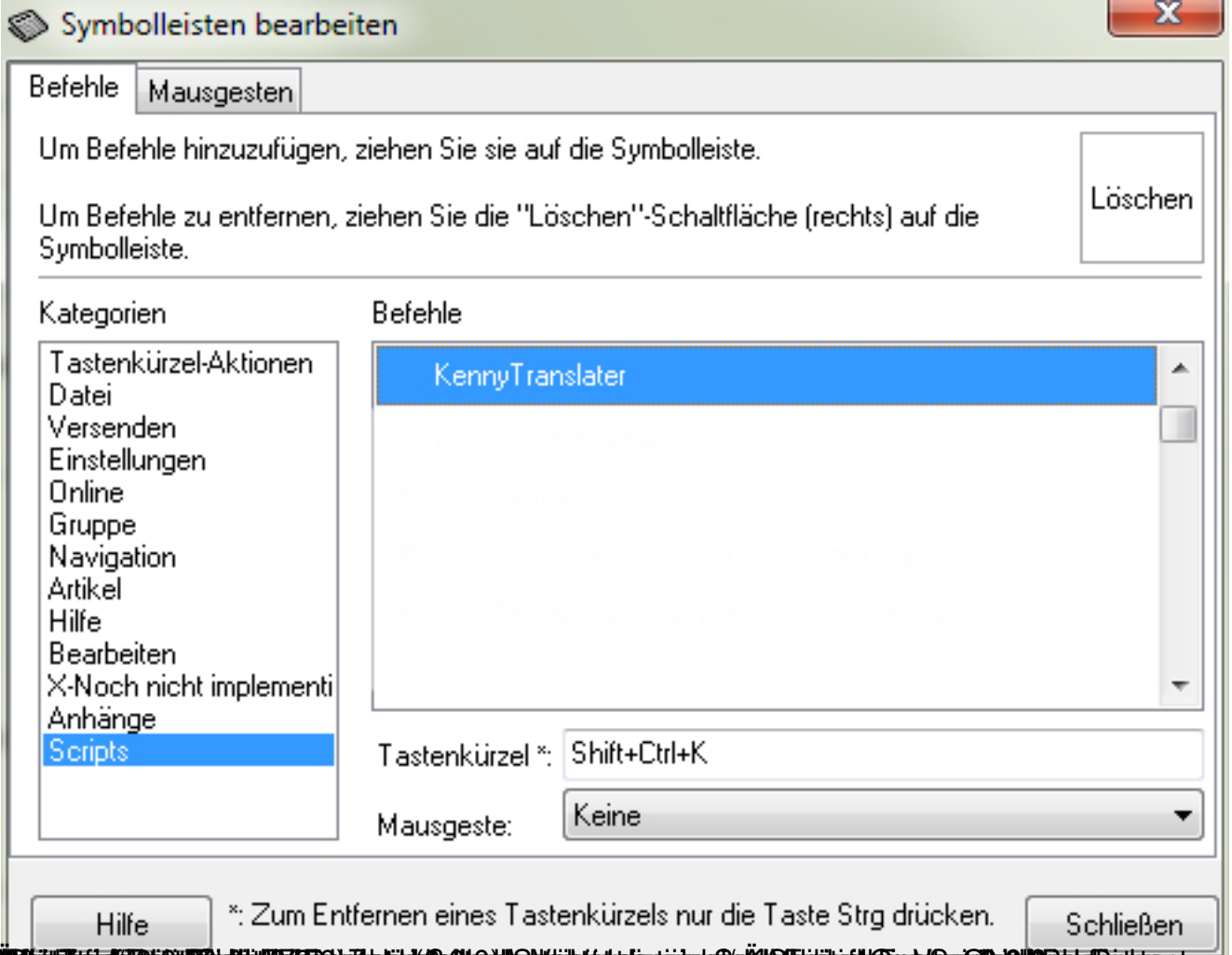
Icon:

Hinweis:

4. Nach einem Klick auf

Zeige Schaltflächenkonfig.

5. kann auch noch ein Hotkey zugewiesen werden



[REDACTED]