Control-Events

Einleitung

In der AutoIt-Hilfe wird ausgeführt, daß mit

```
Opt("GUIOnEventMode", 1)
```

die grundsätzliche Aktivierung des GUI-Event-Modus möglich ist, d.h. dass damit sowohl vom GUI selbst, wie auch von ihren Controls so genannte „Events“ (Ereignisse) generiert werden können, die dann auswertbar sind.
Da in der AutoIt-Hilfe nur sehr wenige Informationen über Details dieser der Windows-Philosophie so grundlegenden Arbeitsweise vorhanden sind, erhebt sich die Frage:
„Was bedeutet das im Einzelnen?“

Die bloße Erwähnung, bzw. die recht dürftigen Hinweise in der AutoIt-Hilfe bringen oft nicht genügend Klarheit, oder man kommt mit der berühmt-berüchtigten „*Try & Error*“-Methode nur mühsam ans Ziel.

Von den 29 in AutoIt verfügbaren Controls sollen deshalb hier die wichtigsten im Zusammenhang mit dem GUI einer eingehenden Untersuchung unterzogen werden.

GUI

Für das GUI sind drei Ereignisse von besonderer Bedeutung, weil sie von den GUI-eigenen Control-Boxen erzeugt werden:

- Schließen - (X)
- GUI-Fenster „herunterklappen“ (auf die Statusleiste) (_)
- GUI-Fenster wiederherstellen (□)

Da bei der globalen Aktivierung des GUI-Event-Modus `GUIGetMsg` nicht verwendet werden kann, müssen in einer speziellen Funktion die dazu notwendigen Aktionen generiert werden:

(nach der Erzeugung des GUI)

```
GUISetOnEvent($GUI_EVENT_CLOSE, "SpecialEvents")
GUISetOnEvent($GUI_EVENT_MINIMIZE, "SpecialEvents")
GUISetOnEvent($GUI_EVENT_RESTORE, "SpecialEvents")
```

Die dazugehörende Funktion sieht dann wie folgt aus:

```
Func SpecialEvents()                        ; hier sind die Special-Events der GUI:
                                            ; Fenster Schliessen, Minimieren, Maximieren,

      ;MsgBox (0, "ID", @GUI_CtrlId)        ; nur zum Test
      Select
        Case @GUI_CtrlId = $GUI_EVENT_CLOSE
            ;MsgBox(0, "Schliessen gedrückt", "ID=" & @GUI_CtrlId & _
                                " WinHandle=" & @GUI_WinHandle)
            Exit

        Case @GUI_CtrlId = $GUI_EVENT_MINIMIZE
            ;MsgBox(0, "Fenster minimiert", "ID=" & @GUI_CtrlId & _
                                " WinHandle=" & @GUI_WinHandle)

        Case @GUI_CtrlId = $GUI_EVENT_RESTORE
            ;MsgBox(0, "Fenster wiederhergestellt", "ID=" & @GUI_CtrlId & _
                                " WinHandle=" & @GUI_WinHandle)

    EndSelect
;
EndFunc
```

Die auskommentierten `MsgBoxen` dienen lediglich Testzwecken, d.h. sie haben sonst keinerlei Funktion.
In der AutoIt-Hilfe gibt es eine Liste mit weiteren Funktionen (`Special-ID`), die hier nicht weiter behandelt werden. Die Event-Funktion auf ein normales `OnClick`-Ereignis fehlt jedoch.
Will man ein normales `OnClick`-Ereignis auf das GUI auswerten, kommt man mit

```
GUISetOnEvent($Form1, "GUI_OnClick")
```

nicht zum Erfolg. Offensichtlich wird dieses „`GUI_OnClick`“-Event nicht unterstützt!
Mit einem Trick ist das aber durchaus möglich.
Dazu wird das Control-Ereignis für `Label` „missbraucht“ (siehe dort).