

**GenCodeOberon**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> GenCodeOberon		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		March 28, 2025	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>GenCodeOberon</b>	<b>1</b>
1.1	GenCodeOberon.guide . . . . .	1
1.2	GenCodeOberon.guide/Wichtig! . . . . .	1
1.3	GenCodeOberon.guide/Anwendung . . . . .	2
1.4	GenCodeOberon.guide/Installation . . . . .	2
1.5	GenCodeOberon.guide/Modulnamen . . . . .	3
1.6	GenCodeOberon.guide/Autor . . . . .	3
1.7	GenCodeOberon.guide/Beispiel . . . . .	4

# Chapter 1

## GenCodeOberon

### 1.1 GenCodeOberon.guide

Dokumentation von GenCodeOberon Release 1.0

\*\*\*\*\*

Das Programm GenCodeOberon ist dafür geeignet, Oberflächen die mit dem Programm MuiBuilder V2.0 erstellt worden sind, in kompilierbaren Oberon-2 Quelltext zu übersetzen.

Die Beschreibung bezieht sich auf Release 1.0, Version 37.1 von GenCodeOberon. Die Anleitung wurde am 18 September 1994 zum letzten Mal bearbeitet.

Unbedingt vor dem ersten Benutzen zu lesen.

Wichtig!      Copyright usw.

Anwendung      Wie man es benutzt

Autor      So erreicht man den Autor.

### 1.2 GenCodeOberon.guide/Wichtig!

Copyright und andere Sachen

\*\*\*\*\*

Copyright (C) 1994 Albert Weinert

Diese Dokumentation darf kopiert und weitergegeben werden, solange die Copyright-Notiz und diese Erlaubnis unverändert auf allen Kopien enthalten sind.

Es wird keine Garantie gegeben, daß die Programme, die in dieser Dokumentation beschrieben werden, 100%ig zuverlässig sind. Sie benutzen diese Programme auf eigene Gefahr. Die Autoren können auf keinen Fall für

irgendwelche Schäden verantwortlich gemacht werden, die durch die Anwendung dieser Programme entstehen.

GenCodeOberon ist Nachrichtenware, d.h. wenn man GenCodeOberon benutzt so muß man dem Autor eine Nachricht schreiben. Darin sollte stehen was man von dem Programm hält, was einem gefällt und was einem mißfällt.

Das Paket ist freely distributable, aber das Copyright liegt weiterhin bei Albert Weinert. Dies bedeutet, daß es von jedem kopiert werden darf, solange er nicht mehr als eine angemessene Kopiergebühr dafür verlangt.

Dieses Paket darf in Public-Domain Sammlungen aufgenommen werden (CD ROM Versionen dieser Sammlung eingeschlossen). Die Distributionsdatei darf in Mailboxsystemen oder auf FTP Servern abgelegt werden. Wenn Sie dieses Paket weitergeben wollen, dann müssen Sie aber die originale Distributionsdatei GenCodeOberon\_V10.lha benutzen. Auch dürfen die Dateien nicht verändert werden.

Dieses Paket darf auch mit dem Programm MUIBuilder zusammen verbreitet werden.

Das Programm und die mitgelieferten Quelltexte dürfen nicht kommerziell verwertet werden.

## 1.3 GenCodeOberon.guide/Anwendung

Wie man GenCodeOberon anwendet

\*\*\*\*\*

Eins vorweg, diese Anleitung ist eine besser wie gar keine Anleitung, sie erläutert wirklich nur das Notwendigste, aber ich denke es müsste reichen.

Installation  
Modulnamen  
Beispiel

## 1.4 GenCodeOberon.guide/Installation

Installation von GenCodeOberon

=====

GenCodeOberon muss zuerst in das Verzeichnis Modules vom MuiBuilder kopiert werden, nur dort wird es von MuiBuilder gefunden. Im MuiBuilder ist nun in den Voreinstellungen Oberon als erzeugten Quelltext vorzubeseetzen. Als GetString sollte in den Voreinstellungen GetString eintragen, den die meisten Programme die aus den catalog descriptions Oberon Quelltext erzeugen stellen diese Prozedur zur Verfügung.

---

## 1.5 GenCodeOberon.guide/Modulnamen

Modulnamen der Quelltexte

=====

Die Modulnamen die von GenCodeOberon erzeugt bzw. von dem erzeugten Quelltext erwartet werden sind folgende.

Bei dem Namen des erzeugten Quelltextes wird immer GUI.mod angehängen, so das in den Code-Options von MuiBuilder in dem Code-Gadget kein .mod angehängen werden darf, der Name der Applikation reicht vollkommen.

Wenn lokalisierter Quelltext erzeugt wird dann muss auch die entsprechende Datei mit den Strings vorhanden sein, dieser Name ist auch vorgegeben. Er wird aus dem Namen des Quelltextes wird einfach ein Strings.mod angehängen. Gültige Lokalisations Quelltexte werden unter andern von FlexCat 1.3 und KitCat 1.1c erzeugt, es mag auch andere geben, aber diese sind mit zur Zeit nicht bekannt.

Da die Möglichkeit besteht Hooks in den Quelltext einzubinden müssen diese auch von dem erzeugten Quelltext importiert werden, damit diese gefunden werden muss der Quelltext der die Hooks beinhaltet aus dem Namen des Quelltextes dem ein Hooks.mod angehängt wird.

Hier nochmal die kurze Zusammenfassung.

Es wird angenommen das der Name des in dem Code-Gadgets eingestellten Quelltextes MuiDemo ist, dann hat von GenCodeOberon erzeugten Quelltext den Namen MuiDemoGUI.mod, das Modul welches für die Lokalisierung importiert wird hat den Namen MuiDemoStrings.mod und wenn Hooks benutzt werden ist der Name des zu importierenden Quelltext MuiDemoHooks.mod.

## 1.6 GenCodeOberon.guide/Autor

Adresse des Autors

\*\*\*\*\*

Falls Verbesserungsvorschläge zum Programm vorhanden sind oder Fehler jedweder Art vorhanden sein sollten bitte ich darum, mich zu kontaktieren.

Dies ginge über die normale Post oder auch Telekom.

Albert Weinert  
Adamsstr. 83  
D-51063 Köln  
Tel: 0221 / 61 31 00  
Deutschland

oder über die elektronische Post, die auf jeden Fall bevorzugt wird.

gencodeoberon%da02@darkness.gun.de

Wenn man mich zu einen anderen Thema als GenCodeOberon anschreiben

möchte, dann kann man dies unter der folgenden Adresse machen.

a.weinert@darkness.gun.de

## 1.7 GenCodeOberon.guide/Beispiel

Example

=====

Hier ist ein Standard gerüst um den erzeugten Quelltext zu verarbeiten.

```
MODULE MuiDemoTest;

IMPORT  Exec,
        Mui,
        gui:= MUIDemoGUI,

        y := SYSTEM;

VAR
    running : BOOLEAN;
    signal : LONGSET;

BEGIN
    running := TRUE;
    IF gui.CreateObjects() THEN;
        gui.CreateNotifys();
        WHILE running DO
            CASE Mui.DOMethod( gui.App,Mui.mApplicationInput,y.ADR( signal ) ) OF
                | Mui.vApplicationReturnIDQuit: running := FALSE;

                ELSE END;
            IF signal # LONGSET THEN y.SETREG( 0, Exec.Wait(signal) ) END;;
        END;

        Mui.DisposeObject( gui.App );
    END;
END MuiDemoTest.
```