oberona

COLLABORATORS

	TITLE : oberona		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		July 22, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1 oberona 1

1.1 GUIEnvironment/OberonA guide 1

1.2 Installing GUIEnvironment for OberonA 1

1.3 The OberonA interface module 2

1.4 GUIEnvSupport for OberonA 2

1.5 Guidelines 2

1.6 Examples for OberonA 2

1.7 rcs 3

Chapter 1

oberona

1.1 GUIEnvironment/OberonA guide

GUIEnvironment

OberonA Guide

==

© 1994 Carsten Ziegeler
Augustin-Wibbelt-Str.7
D-33106 Paderborn
Germany

==

Installation

The interface module

GUIEnvSupport

Guidelines

Examples

1.2 Installing GUIEnvironment for OberonA

To install GUIEnvironment for OberonA simply run the GE_OberonA-Install script !

This script does the following copy actions:

- interface modules (GUIEnv.mod, GEClass.mod and GUIEnvSupport.mod) are stored in the oberon-a:source/3rdparty drawer

```
- examples: oberon-a:Examples/GUIEnvironment

Using the script you now have the choice to compile the interface modules
and after that you can compile and link the examples !


Don’t forget to install GUIEnvironment at first !
```

1.3 The OberonA interface module

```
The OberonA interface module contains all constants, structures and
functions explained in the documents.

The spelling of the constants is slightly different from the C decla-
ration:

- OberonA uses different prefixes without underscore. For example instead
  of GEG_VarAddress a gegVarAddress is used.
```

1.4 GUIEnvSupport for OberonA

```
All functions and definitions are included in GUIEnvSupport as documented
in GUIEnvSupport guide.
Appart from that the GUIEnvSupport module follows the same naming
conventions as explained in the chapter about the interface module !
```

1.5 Guidelines

```
The interface modules offer access to the whole GUIEnvironment without
any exception. So, see the documentation for more information.

But be careful with the callback hooks. As OberonA does not allow register
parameters in any procedure directly, you have to write callback hooks
which use the stack. As the callback hooks are called with registers you
have to use the guiHookInterface tag with the data of the address of the
HookUtil.HookEntry procedure !
This procedure pushes the registers on the stack and then calls your
hook !
But remember to set the guiHookInterface tag BEFORE any other tag handling
with hooks !
```

1.6 Examples for OberonA

```
NotifyExample : Notification of gadtools gadgets & localizing gadgets
                gadget functions and menu functions

ResizingExample: Gadget description flags, font adaptiv
```

```
BOOPSIExample : Resizable BOOPSI gadgets, font adaptiv

GuideExample  : Menu help function & localizing menus

RequesterExample: All requester kinds

TextFieldExample : The BOOPSI textfield gadget
```

1.7 rcs

```
$RCSfile: OberonA.guide $

$Revision: 1.7 $
    $Date: 1994/12/15 15:43:47 $

  GUIEnvironment OberonA Guide

  Copyright © 1994, Carsten Ziegeler
                  Augustin-Wibbelt-Str.7, 33106 Paderborn, Germany
```