# New Menus for Windows 1.4

This software and the accompanying files are distributed "as is" and without any warranties.
The user must assume the entire risk of using the program.

**Please read the file README.WRI!**

**Getting Started**
What you have to read / know

**Introduction**
Introduction to New Menus for Windows Program
About the Author and the Program

Preliminary Remark
Basic functions
The features

**Installation of the Menu**

**Quick Tips**

**Keyboard**

**Special Submenus**

**Internal commands and command line options**

**Bugs, Limits, Help**

## How To Do

**Context-Sensitive Add-Ons for New Menus for Windows**

**The API**

**The Dialogs and Messages**

**The Inifile**


Member of the
Association of Shareware Professionals (ASP)
Doernbergstr.11
D - 34119 Kassel
Voice. (++ 49 561) 780 150
EMail: roger@architektur.uni-kassel.de
CIS: 100412,3624

Fido: 2:2437/501.10

## The Dialogs and Messages

## About the Author and The Program

My first experiences in programming were made with macros in Word and Excel two years ago, at that time I had no idea that I would ever look closer to computers things. (Normally Im busy with the philosophical problems of the Aesthetic of Architecture and Music; by the way, I study Architecture.)
More by accident than by willing, in autumn 1992, I had a contact with Visual-Basic and became amazed with how simple programming for Windows can be.
It was at this time that I first had the idea for a Menu, popped up with the right mouse button, to launch applications - the beginning of my career as a 'leisure time programmer'.

Before the end of 1992 I had created a first version of the Menu in Visual Basic.
This first version supports submenus and Icons (but not with text and Icons).
But the program didnt satisfy me, because it used a lot of system resources.
In spite of how easy Visual Basic might be, it is not usable to create a pop up menu.

The standard in programming seems to be C / C++ (Nothing against Pascal, really!). In spring 1993 I got a C / C++ compiler. Until then I had never heard about Pointers, Classes, sorteable trees, or Callback-functions or Windows-Messages. First I learned the language, and then - because it is boring writing only hello world Programs - I wrote this program: my first and single non-trivial program.

1 1/2 years ago I got an access to Internet, and I looked for other programs, which had the same idea. I realized that the Menu is - seeing it at the historical aspect - a 'Me too' or better still, a 'Me more' program. (Some of these Programs - actually all that I know - have one fundamental problem: They load the Menu and all submenus at the start of the program. If you want to configure several Submenus, they waste a lot of system resources - and it doesn't matter if you have 32 MB RAM installed in your Computer, the System resources are absolutely limited under Windows. My Menu generates the submenu first, when you open the Submenu, and frees all system resources by clicking on the desktop.)

Until now, I assessed my program only as a fingering-exercise to learn C / C++ and the Windows-API, I didn't thought it might be very interesting to the public. But I realized that a lot of other programmers were not wizards also.
I am now developing and supporting New Menus for Windows about the period of one year.
Without the help from many users of New Menus, suggestions for improvements, beta testing, and last not least some error correction in my documentation (See at the bottom of the README.WRI) it would not possible to produce this program. Thanks to all of them!

I hope it's usable for you, and that you have fun with it.

**Used Literature:**

Besides the documentation of Borland C / C++ 3.1 and parts of the Windows SDK I used following literature:


- Schildt, Herbert; C - Befehlsbibliothek; Hamburg 1987 McGraw-Hill Book Company
- Petzold, Charles; Programmierung unter Microsoft Windows 3.1 (Programming Windows); München 1992 Microsoft Press
- Schulman, Andrew; Maxley, David; Pietrek, Matt; Undocumented Windows; Bonn (...) 1992 Addison-Wesley


**'Stolen' Ideas:**

- 'Open Look' by Open Look Foundation
  The Pin to stick Menus on the desktop.
- 'Corel Draw 3' by Corel Corporation
  The Roll Up Button.
- 'Informant' by James S. Straub
  The idea of simulating windows movement in 'real Time' (like NEXT). I improved this simulation

through a Message Loop in WM_MOUSEMOVE to restore the uncovered Regions. But it is still flickering.

## Preliminary Remark

Why should you use  this program?

The Program Manager of Windows has several aspects, which can't satisfy the (power-?)user.

- You can't create subgroups in the Program Groups.
- Often you have to fish the Program Manager out of the overcrowded Desktop.
- It doesn't seem to be worth to install Documents of your actual Projects in the Program Manager.

Enough of lamentations! How Ive tried to compensate this flaws, see: Basic function of the Menu

## Basic functions of the Menu

Some other Graphic User Interfaces dont have a Program Manager, as known by Windows. They use a menu to launch programs. This menu normally appears by clicking on the desktop.
The model for my Menu is the Desktop Menu of OpenLook (which you may recognize by the pin).
After popping up the Menu with the right mouse button, you can start programs or open documents.
You can group all the programs in submenus in a similar way to the Program Groups of the Program Manager. Often used programs and documents in Submenus and seldom used programs and documents in SubSubSub...Menus.

For an overview of the important features of the Menu see: The features of the Menu.

## The features of the Menu

- Only 2 % - 3 % of system resources are used permanently (Program Manager ~ 4 - 5 %).
- Only ~40 KB of non-discardable RAM needed permanently (Program Manager ~16 KB). But in the time of virtual memory this might be more theoretical. If there is enough RAM, the Menu allocates ~140 KB, the Program Manager ~190 KB).
- A freely customizable Menu, with as many Submenus as you like.
- Supports the right as well as the middle mousebutton.
- You can also use the Menu without a mouse.)
- Shows automatically the icons of programs and documents in the menus in 3 possible sizes.
- Freely customizable menu font.
- Freely customizable menu colors.
- Freely customizable menu icons in three sizes.
- ShortCuts for Submenus
- HotKeys to launch programs

- Supports following generic Menu items
  - Supports ambiguous filenames (like `c:\mydir\*.txt`), so you can list all existent files matching a Searching Directory
  - You can include a small generic 'File Manager' in your Menu, which shows all subdirectories of your hard disk as submenus and all programs and documents as items.
  - You can include the Program Manager or single Program Groups in your Menu.
  - Submenus, which are too long to fit the desktop, will break and be continued in a new Submenu called More... (It was very hard to do this. The obvious things are the most complicated.)
  - A Task-List, to switch quickly to an application.

- You can execute whole submenus as batches

- You can stick the Menus with the pin on the desktop.
  - It is possible to hold the sticked Menus on the top of the desktop
  - Minimize the sticked Menus with Roll-Up-Button.
  - Supports Drag and Drop with the File Manager to install Files as Menu items in a sticked Menu.
  - Supports Drag and Drop with the File Manager to open any File with a program, installed in a sticked Menu.
- You can also stick normal Drop-Down Menus of Win-programs with a pin.

- Context-Sensitive Menus enhances the functionality of Windows controls.
  For File-Manager, Notpad, and DOS-Boxes there are predefined context sensitive Menus.

  A integrated Virtual Desktop

## Installing the Menu

## Requirements

The Menu needs following Hard- and Software:

- Windows 3.1
- A mouse with 2 or (better) 3 buttons
  (It also functions without a mouse)
  Note: If you use a mouse with three mouse buttons, and the middle mouse button is not supported, try another mouse driver. As far as I know Microsofts mouse drivers don't support the middle mousebutton. Try the mouse driver "System Mouse" or "Serial Mouse". Maybe also the mousedriver from Genius mice work.
- A hard disk cache (like smartdrv) is highly recommended.

## Installation of the Menu

For installation and deinstallation please look in the README.WRI.

Lock also: Install the Menu in a Network and The Files

## The Files

Please refer to README.WRI for the actual list of the files.

The following files may created in the directory where RRKMENU.EXE was installed:
RRKDAT.xxx
RRKIDX.xxx

If you use the feature to show the program and document Icons, the Menu extracts the Icons from the Programs and write this information in two database files.

The file extensions "xxx" are named after a combination of the Icon size and the color resolution of your graphic driver.
If you use a color resolution with two colors (black and white) the Bitmap-file needs 1 Bit for each pixel (.01x) 16 colors needs 4 Bits (.04x) and 256 colors etc. 8 bit (.08x) etc.
If You use an icon size of 16 Pixel the third 'x' is a 1 (.xx1); by using an icon size of 24 Pixel a 2 (.xx2) and for 32 Pixel a 3 (.xx3).

The file RRKIDX.xxx is the index file for RRKDAT.xxx. It list the offsets of the Icons stored in the RRKDAT.xxx (Not really a Windows-Icon, nor a Windows- Bitmap, more a Core-Dump of a Bitmap, because real Bitmaps would be to slow. This causes the different xxx's. If you use the 32 Pixel Icon size with a Color resolution of 16 million colors this file (RRKDAT.243) may become huge.)

## Install the Menu in a Net environment

If you want to install the Menu on a File-Server, you can declare the directory of the RRKMENU.INI as the argument of RRKMENU.EXE

Supposing the File-Server is associated with `d:\` and the Menu is installed at `d:\windows\rrkmenu` and the local users hard disks with the users INI-files is associated with `c:\`, so you can install the Menu in the Program Manager with the command line `d:\windows\rrkmenu\rrkmenu.exe c:\windows\rrkmenu`
In this directory `c:\windows\rrkmenu` the Menu will also create the Files RRKDAT.* and RRKIDX.* for the Icons database.

If you want to use the Menu as the windows shell, you can't declare shell=`d:\windows\menu\rrkmenu.exe -menupath c:\my_dir\menu` in the SYSTEM.INI, because Windows doesn't support arguments in the shell= statement.
You can handle this problem with a batch file.
First you can define the Menu as Shell in the Dialog **General Properties** in the section **Miscellaneous** with the option **As Shell**. After that you can create a batch file e.g. WINMENU.BAT:

```
win -menupath c:\my_dir\menu %1 %2 %3 %4 %5 %6 %7 %8
```

This way you also can create different batch files for different users (or uses).

If you want to install the Menu for several users on different computers, it might be desirable to share parts of the menu structure with all others users, which can't be modified by the user. Then it is possible to define external Submenus.

## Document

A Document is a file associated with a program, identified by the file-extension.
For example: A File with the file extension `.wri` is a Write - Document.

You can associate Files with Programs in the dialog Dokument -Association.

## Pinned Menus

You can stick the submenus on the desktop by clicking on the green Pin. In fact, it isn't really a Menu, but it acts like one.
In contrast to normal Menus, the pinned Menus don't disappear by clicking on the desktop. You can move the pinned Menus by dragging them by the top; You can roll them up and down with the red / green arrow button; With another click on the pin you can hold the Menu at the top of the desktop and you can close it with a 3rd click on the pin.

## More ...

The menus support an automatic 'menu break'. If a submenu tends to become too long, at the end of the menu a submenu called **more ...** is appended. This Submenu contains the rest of the Menu items.

## Highlight Menuitems to edit a Menuitem

**In normal, non-pinned Menus**

- If you want to edit a Menu item, first you have to pop up the Submenu, by pressing the right (respectively the middle) Mouse button.
- Then you have to highlight the Menu item with the left Mouse button.
- **While** highlighting the Menu item, you can use one of the Edit keys (e.g.: F5).
- After the Menu has disappeared a dialog appears; you can release the left mouse button.

**In pinned Menus**

*Either:*

- Highlight a Menu item.
- Press one of the Edit keys (e.g.: F5).

*or:*

- In pinned Menus you only have to click with the right (respectively with middle) Mouse button on the Menu item to edit.
- In the popped up Edit Menu you can choose an edit option.

## Default Icons

 General program (Windows or DOS), without its own icon.

 Document of a program, associated with a program, without its own icon.

 A COM-Program (DOS-Program).

 A DOS-Batch-File.

 A DOS-Program with a PIF-file.

### Popup the Menu

- If you have selected the option **only Desktop** in the Submenu **Menu Setup**, you must move the mouse cursor to a place, where the background of the desktop is visible.
- Push the right (respectively the middle) Mouse button.

*or:*

- Push the **Alt** and the **Pause** - key.

**Close the Menu:**

- Click anywhere on the desktop.

*or:*

- Push one ore several times the **Esc**-key.

**Menu item**

A Menu conventional item contains a command like 'File - Open'. In the Menu, normally, a Program will be started by selecting a Menu item.
Some Menu items contain submenus (cascading Menus).

**Iconfile**

Files with the file extension ICO (sometimes also ICN) contain one Windows icon. Often, files with the file extension DLL or EXE also contain one or more Icons.
Some icons can be found in the files PROGMAN.EXE and MORICONS.DLL in the Windows directory.

## Drop-Down Menus

Drop-Down Menus are the normal Menus of a Windows program with a menu bar (File, Edit, and so on). If you open a Drop-Down Menu with the right mouse button, you get a graphic Menu with a pin to stick it on the desktop.

See also: Stick normal Drop-Down-Menus with a Pin.

### Checked Menu items

In normal Windows menus, checked items have a little check mark at their left side.
In Menu, checked Menu items are shown in different ways depending on the chosen style (look).
If you selected the OpenLook style checked Menu items are identified by underlining their description. In other styles it depends, on whether the menu item has an icon. If the menu item has an icon, the icon of a checked menu item has a border (similar to a button). Otherwise the menu item has a little check mark.

### First opening of a Submenu

Submenus are initialized the first time they are opened. This means the information for the submenus is read out of the RRKMENU.INI, or calculated in the appropriate way (e.g. in the Submenu "Tasks" the running Tasks are listed etc).
This initialization is also important if you want to convert GRP-Files to the internal format.
No matter how many times you pop up the Menu, its submenus have been initalized only once. You may refresh pinned menus using the **F12**-key.

## Stick Menus on the Desktop

- Click in a popped up (Sub-)Menu on the green pin.

- To hold the pinned Menu always on the top of the Desktop, click a second time on the green pin, or press the **Shift - Esc** - keys.
- To close the pinned Menu, click a third time on the green pin, or press the **Shift - Esc** - keys.

**Execute a Menuitem of a pinned Menu**
A click on a Menuitem will start the corresponding Program will be start, respectively the Submenu will be open. In contrast to normal Menus a Submenu will first open, if the mouse button is released.
If you press the **Shift**-key while clicking on the Menuitem, the pinned Menu will be rolled up after executing the command line.
If a Menuitem in a the pinned Menu is already marked you only have to double-click on the Title bar of the pinned Menu. This also works, if the Menuitem is a Submenu, or if the pinned Menu is rolled up.

## move the pinned Menus

- To move a pinned Menu, drag the Menu with the left mouse button on the Headline of the Menu, or use the **Shift** and the **arrow**-keys.

See also: Arrange all pinned Menus

## MINIMIZE the Pinned Menus

- Click with the left mouse button on the Roll-Up-Button.

*or:*

- Push the **Shift** and the **PgnDown** - Key

**To restore the size of pinned Menu:**

- Click with the left mouse button on the Roll-Up-Button.

*or:*

- Push the **Shift** and the **PgnUp** - Key

See also: Arrange all pinned Menus

## Arrange all pinned Menu on the Desktop

If you want to arrange all pinned Menus - excepting the Menus with icons only and the pinned Drop-Down Menus - in lines at the top of the desktop:

- select in the local Editmenu **arrange at top**,

*or:*

- push the **F11-**key.

If you want to arrange all pinned Menus - excepting the Menus with icons only and the pinned Drop-Down Menus - in columns at the left of the desktop:

- select in the local Editmenu **arrange at left**,

*or:*

- push the **F9-**key.

## Hold pinned Menus on the Top of the Desktop

It might be desirable to hold a pinned Menu every time in the foreground of the Desktop.
To do that, click with the left mouse button on the pin, so the pin looks like: 
You can also use the **Shift-** and **Esc**-key.

If you click once more on the pin, the Menu will be close.

## Refresh the pinned Menus

In different cases it is useful to refresh a pinned Menu:

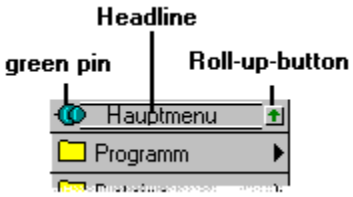
- If you changed the icon- or font size in the Dialog for changing general Properties
- If you have a **Searching Directory** in your pinned Menu, and you create or delete a File.
- Last but unfortunately not least, if there is nonsense in the Menu, or the Menu becomes too long.

To refresh a pinned Menu:

- activate the Menu (there are two thin lines showing, when the pinned Menu has the focus).
- And push the **F12**-key.

The Menu might disappear, after pushing F12-Key. In this case, the pinned Menu contains a Drive Directory.

## The title bar of a pinned Menu



## Elements of the INI-file

Entries in the WIN.INI

The INI-file - General Properties

Submenus in the INI-File

Syntax of a Menu item in the INI-file

Entries in the VSCREEN.INI

## Entries in the WIN.INI & REG.DAT

The New Menus for Windows doesn't remark its existence in the WIN.INI.
If you want deinstall NMfW you can delete the version information in the section "New Menus for Windows" in the REG.DAT using REGEDIT.EXE.

## The INI-file - General Properties

This chapter documents the switches for General Properties in the RRKMENU.INI

**Section [Language]**

FirstStart=0

After starting the Menu the first time, it is 0

German=0

1 = German dialogs and help file are used;

0 = English dialogs and help file are used.

(Don't change it manually, it's better to reinstall the New Menus for Windows)

ShowInitScreen=1

0=OFF; 1=ON

Shows the startup screen.

**Section [General Properties]**

*Item=default*

AsShell=0

0 = Off; 1 = On.
Using the Menu as Shell.
(Don't change this manually, it wouldn't work!)

ShowEnhanced=1

0 = Off; 1 = ON.
Shows the Menu items with a 3-D-effect and customizable Font.

OWL_LIKE=1

0 Standard; 1 OpenLook; 3 beta.

WithIcons=1

0 = Off; 1 = ON.
Shows the Program Icons in the Menu.

IconSize=16

16, 24 or 32
The Icon size in the Menus.

IDXinRAM=1

0 = Off; 1 = On.
Hold the index file RRKIDX.xxx and parts of the REG.DAT in RAM.

mMouseBotton=0

0 = right Mouse button; 1 = middle Mouse button.
The Mouse button to pop up the Menu.

OnlyDesktop=0

0 = Off; 1 = On.
On: The Menu only will pop up, when the cursor is over the background of the desktop

NoClient=1

0 = Off; 1 = On.
If OnlyDesktop=1 and NoClient=1 the menu also can be popped up over a title bar of a window.

IconVisible=1

0 = Off; 1 = On.
Shows or hides the program icon on the desktop (the black arrow).

SaveDT=1

0 = Off; 1 = On.
Saves the sticked menus position on the desktop when quitting and restores them by restarting the Menu.

KeySub=1

0 = Off; 1 = On.
If ON you can use **Alt + Pause** and hotkeys to pop up the menus.

HookMenus=1

0 = Off; 1 = On.
Normal Drop-Down-Menus of a window application are selectable with the right mouse button.

ClassHook=1

0 = Off; 1 = On.
Context Sensitive Menus are available.

RealMove=1

0 = Off; 1 = On.
Move the sticked Menu in 'Real-time' as in the NeXT desktop.

mnu_sdt=1

0 = Off; 1 = On.
Only available if the Menu is the Windows Shell.
Controls if a menu item in the Submenu **Tasks** called **Save Wins DT** is appended. With this Menu item you can save the positions of all Windows on the desktop. If the Menu (and Windows) restart, all Windows (programs) are restored. Unfortunately the documents can't be reloaded.
If you press **Ctrl** while selecting this Menu item, all positions will be 'deleted'.

SpezialRightMouseButton=1|Tasks|MNU_TASKS

Defines a Submenu you can pop up with the right mouse button over the background of the desktop or over the title bar of a window.
You can also declare a program (e.g. "0|Editor|NOTEPAD.EXE").
This only works, if you normally use the middle mouse button to pop up the Menu.

OldColorNum=0

It's just to remind the Program Manager to restore its icons if you changed the number of colors. (Don't change this by manually!!!!)

FileViewer=NOTEPAD.EXE
FileEditor=NOTEPAD.EXE

Declares programs, which are used, if you select files which arent associated with any application.

SystemMenu=SystemMenu

Name of a Submenu, to be appended to the System menu.

MaxHistory=20

Max. length of the "History" submenu.

HistorywintCommand=0

0=Off; 1=On
If ON, the the internal menu commands will also be stored in the "History" submenu.

SysMenuAlsoInTitle=1

0=Off; 1=On
If you click with the right mouse button on the title bar of a window, you get the system menu.

MMouseEmu=3

1=left + right mouse button
2=Ctrl + right mouse button.
With this combinations you can emulate the middle mouse button.

IsDatFileClosed=1

0=Off; 1=On
Only used for debugging.

CheckFileBase=0

0=Off; 1=On
Only used for debugging.

NoConfirm=1

0=Off; 1=On
Quit the Menu without confirmation.

DynamicPMG=0

0=Off; 1=On
The Progman-Manager groups are built dynamically. Otherwise the groups are converted to the MNU-format and stored in the PROGMAN.MNU

IncludeHiddenTasks=1

0=Off; 1=On
If ON, the hidden windows will also be listed by default In the "Tasks" submenu. In the System menu, the item "Hide" will be available.

WithTitleLabel=1

0=Off; 1=On
The label of a Submenu also appears on the menu title bar.

TaskWithAllWindows=0

0=Off; 1=On

If On, shows all (popup) windows of a program. If Off, shows only the main window of a program in the submenu Tasks.

MainMenu=1|MainMenu|RRKMENU.INI

Defines the MainMenu, which pops up if the right or middle mouse button is pressed.

NTCompatible=0

0=Off; 1=On

If On, all undocumented functions are disabled.

MoveIconsAllways=1

0=Off; 1=On

If On, all icon sized windows stays in the visible Virtual Desktop.

WithGDICache=1

0=Off; 1=On

Reserves some graphical Object from Windows. With this option the menus will be showed faster.

WithVScreen=1

0=Off; 1=On

The Virtual Desktop is on/off

XMouse=0

0=deactivated

1=activates the window under the mouse cursor, but the order of the window stack will be hold

5=activates the window under the mouse cursor and the window will be pops up into the front.

PopupAllign=4

0=Left

4=Middle

8=Right

The allignmenent of a poping up window to the mouse cursor.

### Section [MenuFont] and [MenuTitleFont]

The sections [MenuFont] and [MenuTitleFont] has the same structure.

Name=MS Sans Serif
Size=-11
Weight=400
Width=0
Italic=0
Underlined=0
Align[M|T]Font=0

(Don't change this option manually)

### Section [Autostart] or [Startup]

Is a normal Submenu, excepting that the menu items are executing when Windows is starting.
The Menu has to run as Shell.

**Section [ShortCuts]**

A=1|Tasks|MNU_TASKS<0

Ctrl - Char=INI-Statement for a Submenu.

These are the shortcuts to pop up the submenus

You can use the dialog **General Properties / ShortCuts** to edit this items

**Section [ProgrammHooks]**

196689=0|Quit|MNU_END<0

LONG-Number=INI-Statement

List of the Hotkeys.

LONG-number: KeyCode .

INI-Statement: A menu items INI-line.

You can use the dialog **General Properties / Hotkeys** to edit this items

**Section [WinClasses]**

Edit=1|EditHook|ADDONS.MNU<10

ClassName=INI-Statement

A list of all registered window classes, which have a context sensitive Menu.

If you want totally exclude a window class from supporting context sensitive menus or popup menus, because the program supports its own right mouse function, use MyWinClass=Excluded.

You can use the tool EXTRMNU.EXE to find out the class name of a program.

### Section [Menu_Desktop]

List of all sticked Menus, which are restored, when the Menu restarts.

### Section [Wins_DT]

Stores a list of open applications windows and their coordinates.

## Submenus in the INI-File

Don't delete the sections **[MainMenu]**, **[Menu Setup]**

This chapter documents the entries for submenus and menu items in the RRKMENU.INI. You don't need to edit the RRKMENU.INI manually; all sections are editable using the dialogs. Nevertheless, I think its important to document the important files.

The RRKMENU.INI has the normal format of Windows INI files. Using my own format might improve the speed of Menu for a few milliseconds, but the disadvantage of human unreadable file format isn't worth it. Each submenu is marked with brackets "[" "]". Below the Menu items are listed.

Following Submenus are reserved:

[MainMenu]
: The root (Sub)Menu.

[Menu Setup]
: Some items to configure the Menu.

[Autostart] or [Startup]
: When running the Menu as Windows Shell, all Items listed here will be started at the beginning of Windows. (You can create this submenu in the same way as other submenus.)

[Wins_DT]
: The Programs which saved with the command MNU_SDT.

[WINCLASSES]
: All hooked Windows-Classes are listed here.

[ProgrammHooks]
: The hooks to command lines.

[Menu_Desktop]
: List of the menus sticked on the desktop, to restore at startup.

**Example:**

```
[My Submenu]
;          The name of the Submenu.
0=0
1=0|Paintbrush|pbrush.exe
2=1|Files|NIX
3=2|*.DOC|C:\D\C\MENU\*.DOC
4=8|c:\windows\|c:\windows
```

**Characteristics of a Submenu:**

The first entry 0=`n` is not a Menu item, but a number to describe the characteristics of a Submenu. If the entry 0=`n` is missing, the default for `n` is 0;

The number `n` is interpreted bit-wise:

- 1: The Submenu is showed only with the Icons.
- 2: The Submenu is showed without the pin to stick the Submenu on the desktop.
- 4: The Submenu is showed without the Icons, only with the text.
- 8: The Submenu is locked and thus is not editable with dialogs.

To get combinations of this characteristics just add the numbers.

## Syntax of a MENUITEM

MenuitemNumber=Typ of Menuitem|Label|File [Arguments][>Working Directory][<Properties][;Iconfile Iconpos]
[] = optional

- MenuitemNumber: individual Number in the order of the Menu items
- Type of Menu item:
  - 0: normal Menu item
    The **Label** is shown in the Menu and the **File** will be executed
  - 1: Submenu
    The Label is shown in the Menu as a submenu. It links to another Submenu. The File is a dummy (NIX == Nothing), generated by the Menu.
  - 2: Searching Directory
    The Label is a dummy, generated by the Menu. All Files which are matches to **File** are listed in the Submenu.
  - 8: Drive Directory
    The **Label** is shown in the Menu as a Submenu. The **File** is the starting directory; all subdirectories of this starting directory generate new Submenus, all Programs and Documents are shown as Menu items.
  - 128: vertical Dividing Line. **Label** "-" and **Filename** "x-"; x is an individual number.
  - 256: horizontal Dividing-Line **Label** "" and **Filename** "x"; x is an individual number.
  - 8193: Program Groups
    Creates a submenu with the Programs installed in a Windows Program Manager Group.
  - Modifier (Add the following values to the value of Drive Directory or Searching Directory to modify their behavior):
    - 65536: List all files, including the files, which aren't documents nor programs
    - 131072: List files sorted by file extension.
    - 2097152: List files sorted by date.
    - 4194304: List files sorted by size.
    - 8388608: List files in reverse order.

- Label: see **Type of Menu item**
- File: see **Type of Menu item**
  *reserved expressions:*
  For more information, look in Internal Commands
- Arguments:
  The arguments of a Program
  Only used if the Type of Menu item == 0
  Default is no argument
- Working Directory
  default is the directory of the program
- Properties:
  The size of starting Program
  0 = hidden;
  1=normal;
  2=minimized;
  3=maximized
  Only used if the Type of Menu item == 0
  (Default is the normal)

  If the Menu item contains a Submenu the meaning of Properties is:
  - 1: The Submenu is shown only with the Icons.
  - 2: The Submenu is shown without the pin to stick the submenu on the desktop.
  - 4: The Submenu is shown without the Icons, only with the text.
  - 8: The Submenu is locked and thus is not editable with dialogs.

  (Default is 0= with Icons and text)
  To get combinations of this characteristics just add the numbers.

- Iconfile Iconpos:
  If the icon of the menu item is explicit declared, this represents the file which contains the Icon and the IconID. You can also declare a second Icon in the same Iconfile for the highlighted state.

Comment: "Unsinn" is German and means nonsense. Ignore this entries.
"NIX" is a short German word for "nothing" and is used for the filenames in submenus. In versions > 1.32 this keywords aren't used longer.

## Entries in the VSCREEN.INI

[Window]
top=5
left=5
width=113
hight=90
    Position of the Virtual Desktop window
WithCaption=0
    0=False
    1=True
    With title bar.
rows=2
cols=2
    Numbers of columns and rows in the Virtual Desktop window
Sticked=0
    0=False
    1=True
    The window of the Virtual Desktop stays at the top of the window stack
SwitchToActivWindow=1
    0=False
    1=True
    If a window received the focus and this window stays in a not in the current virtual desktop, the virtual desktop will activated
Minimized=0
    0=False
    1=True
    The window of the Virtual Desktop is minimized.

## Restart Windows

**Yes**

Windows will end and then restart.
If you have chosen the Option **As Shell**, instead of the Windows Program Manager the Menu will start at the beginning of Windows.

**No**

Windows will not restart.
If you have choose the Option **As Shell**, instead of the Program Manager of Windows the Menu will start at the next beginning of Windows.

**Cancel**

Your changes for the option **As Shell** will be discarded.

## Sorry; Drive directories, Searching Directories and PM-Groups can't be edited in pinned Menus. Use the Edit-Keys.

You can't edit Drive Directories, Searching Directories and PM-Groups in pinned Menus. To change the Menuitem popup the Menu, highlight the Menuitem and use the Edit keys.

## There is no Menuitem selected!

You have to select a valid Menuitem, which you want to edit. Use the right (middle) mouse button and click on the Menuitem to edit.

## File-Error - File can't be opened.

Restart Windows to close all files.

## A Hint: You can only edit the beginning of a Drive Directory!

You highlighted a Menuitem contained in a Drive Directory.
If this Menuitem is not the beginning of the Drive Directory, changes won't take effect.

## You have to declare a Label and a File!

See: The Edit Dialog

## You useD a reserved expression!

The expression as filename "mnu_"... is reserved. You can't edit Menuitems, containing this expression.

## Attention! Do you wish to keep the old Submenu?

If you edit (change) a Menuitem containing a Submenu, you have to choose whether the old Submenu should be deleted or preserved.

## Exit Windows?

**Yes:**
Exit Windows.

**No:**
Neither Windows nor the Menu will be ended.

**This File-Extension is already associated with a Program! Before making a new Association, detach the old one.**

See: The Dialog Document - Association

## You have to declare a Program-File and a File-Extension!

See: The Dialog Document - Association

## The declared Association is not present!

See: The Dialog Document - Association

# Bugs, Limits, Help - Index

## Known Bugs

*Problem:*
If a DOS-box (or a program which is running in a DOS-box) is activated, editkeys, hotkeys, shortcuts, and the shift-deactivation for the mouse don't function.
*Solve:*
Sorry, there is no solution available. The Menu can't hook the keyboard, if a DOS-Box is activated. If you want to use editkeys, hotkeys or shortcuts bring another window to the top. If you want to use the context-senstive right mouse button for DOS-Boxes, comment the line
;tty=1|DOS TTY|ADDONS.MNU<10
(write the char ';')
in the section [WinClasses] of the RRKMENU.INI.

*Problem:*
If you are too fast while selecting Submenus in pinned Menus, it can ignore your mouse-action.
*Solve:*
Keep calm, do the same, but a little bit slower.

*Problem:*
Sometimes (maybe depending on the mouse-driver) the Menu annoys you, popping up, although you don't press the right (middle) Mousebutton.
*Solve:*
Press the left and after that once more the right Mousebutton.

*Problem:*
Seldom the Menu can corrupt other normal menus. If you recognise that other menus are changed in an abnormal way, you must restart Windows.
There is a serious bug in the Windows-Management of the menus. (CreatePopupMenu(), AppendMenu(...) and DeleteMenu(...) seem to fail sometimes in the depth of the Windows-API by stressing out of the normal efforts. Sometimes calling CreatePopupMenu() I even received the HMENU from the System-Menu!!! (That's a Bug!!!(?)) Any idea?)

*Problem*:
The **vertical Dividing Line** does not function in a desirable way. Especially if you stick the Menu, containing a **vertical Dividing Line** the pinned Menu can become too long.
*Solve*:
In this version of New Menus for Windows - if you have not fallen in love with this option - avoid it.

*Problem*:
Some Graphic Boards (e.g. Spea V7) don't draw the border for the new position of the window if you move a pinned menu.
*Solve*:
Use the option **RealMove** in the section **Miscellaneous** of the dialog **General Properties**.

## It doesn't WORK - Help!!

***Problem***:

The Menu doesn't pop up by pressing the right Mousebutton.

Try it with the middle mouse button or deactivate the Option middle Mousebutton

***or***

You marked the option Only Desktop

Move the Cursor over the Background of the desktop and try it again.
or:
Deactivate the option **Only Desktop**.

***Problem:***

If you start RRKMENU.EXE an error message appears:
*"Can't find BWCCxxx.DLL"*

This error message is not so serious, like it sounds. All I know it is caused by a bug in older version of the file BWCC.DLL.
Look at your hard disk, if there are any old versions of the file BWCC.DLL and deletes them. Put into your \windows\system directory the newest version of the file BWCC.DLL.

***Problem:***

If you start RRKMENU.EXE an error message occurs like:
*"Undefined dynalink in RRKMENU.EXE"*

This error message is probably happened, because there are old versions of the file RRKMNDLL.DLL onto your hard disk.
Look if you can find any old version of this file - especially in the directory \windows\ - and deletes them, and copy the new version from your \nmfw directory into the \windows\system directory.

## Internal Limits

- Shown Menus and Submenus at one time:
  If you reach the limit of the System resources a Beep will be sound.
  (The Menu frees the System resources, if you click on the Desktop.)
- Lists only the first 1000 Menuitems in a sorted **Searching Directory** or **Drive Directory** (only including programs and documents).
- Maximum of Submenus and Menuitems
  The file RRKMENU.INI must not be bigger than 32 KB.
- Registered (and shown) Icons in the Database file RRKIDX.xxx
  The file RRKIDX.xxx must not be bigger than 32 KB.
- Pinned Menus
  50

## !!BEEP??

The New Menus for Windows sounds in the following cases:

**You selected a Menuitem and execute it**,
The Program or document can't be found.

**You have selected a Submenu**
You reached the limits of the system resources
The Menu frees the System resources, if you click on the Desktop.

**Otherwise:**
A general trapped fault - I hope it is not serious.

## Quick Tips

## What is it?

The New Menus for Windows are a set of Popup Menus, in which you can install programs and documents to launch.

If you push anywhere the right mouse button, this Menu will be popped up at your mouse cursor position.

## Call the Menu

Press the **right mouse button**!

**To close the popped up Menu**
Click anywhere with the left mouse button on the desktop.
If you push on the green pin, you can stick the Menu on the desktop. If you want to get rid of this pinned Menu, click two times more on the Pin.

## Ending The Program

- Popup the Menu and select **End** in the Submenu **Menu Setup**.

*or:*

- Select **Close** in the System menu of the program-icon (a black arrow).

## Launch Programs

The Menu is not configured for your needs.
The Menu is only configured initially with a few Menuitems. I assumed, your Windows is installed at `c:\windows`. In the case my hypothesis is correct, you can access the Menuitems in the Submenu **WinPrograms**. Otherwise you have to change these Menuitems.
I also assumed, you have a `c:\` disk.
(But this configuration really doesn't exhaust the possibilities of the program.)

If you select a Menuitem in the Submenus or Sub-Submenus **WinPrograms**, **C:** or **Progr. Manager** normally a program will start or a Document will open.

## Configure the Menu

Because the wishes of the users and also the possibilities of this program are manifold, I give here only a short hint:

If you want costumize your menu structure or if you want to change general properties, select the menu item **Configuration**. The Dialog **Menu - Configuration** is the central place to edit menu items or do other customizing.

See also Keyboard for more Edit keys
and read the chapters of How to do in the Index window.

## Special Submenus

Tasks
Setup
Window-Manager
Virtual Desktop
Systemmenu
Printer
History
Control Panel
Context-Sensitive Menus
Systemmenu of pinned Menus
Editmenu of pinned Menus
Startup = submenu Autostart

## Submenu Tasks

(1|Tasks|MNU_TASKS)

All visible windows (Programs) are listed in the order of theire appearance.
If you enable the option **with hidden Tasks** in section **Miscellaneous** of the dialog **General Properties**, you get also the invisible Tasks (windows) listed.
If you hold down the **Shift**-key while opening the Submenu **Tasks**, also the invisible windows will listed, independent of the **with hidden Tasks** option.

The invisible tasks are checked. If you click on a invisible tasks in the list, the corresponding window will be showed. The other way, you can hide a window, if you select a task in the list while hold down the **Shift**-key.

By selecting a Menuitem the window will be bring in to the top of the desktop. But if you press the **Ctrl**-key while selecting a Task, the corresponding Program will be quit, respectively the corresponding window will be closed.

At the bottom of the task list you can also see the current system resources:
Free Memory and
Free Resources (GDI = resources for graphical objects, USER= resources for window-objects).

## Setup
(1|Setup|MNU_SETUP)

Run
Only Desktop
Hide Icon
Configuration
Info
End
Reboot

## Run

(0|Run|MNU_RUN)

This Menuitem prompts you for a commandline.
If you just want to modifiy an command of an existent menu item, press the Ctrl-key while executing this menu item.

See also: Dialog Menu - Execute

## Only Desktop

(0|Only Desktop|MNU_BACKGROUND)
If you want, that the Menu will only popped-up, when the mousecursor is over the background of the Desktop, you can select **only Desktop** in the Submenu **Menu Setup**.


1. The Menu will popped up on every place of the desktop.


2. The Menu will only popped-up, if the mousecursor is over the background of the desktop.


3. The Menu will only popped up, if the mousecursor is over the background of the desktop **or** if the mousecursor if over a title bar of a window

You can toggle between 2 and 3 with pressing **Ctrl** while selecting **only Desktop**.

See also: Avoid Conflicts with other Programs.

**Hide Icon**
(0|Hide Icon|MNU_HIDEICON)

 Visible.

 Invisible.

One of the aim of the Menu is to clean up the overcrowded Desktop. By consequence you can also hide the program icon of the Menu (a black arrow).
Normally the program-icon is showed at the bottom of the screen. If don't want it on your desktop, hide it.

### Info
(0|Info|MNU_INFO)

Shows the program information.

## End

(0|End|MNU_END)

Ends the Menu.
If the Menu is running as Windows Shell or if you pressing the **Ctrl**-key while selecting End, after confirming Windows will shoot down.
Alternatively you can shoot down Windows with the command line
MNU_END 1
If you haven't activated the option **Confirm End** in the section **Miscellaneous** of the dialog **General Properties** Windows shoot down without a confirmation. You can also skip the confirmation if you press the **Shift**-key, while selecting this menuitem.
See also: Running a Windows incompatible DOS Program

## Reboot

**Quit Windows**
(0|Quit Windows|MNU_END 1)
Does that, what it tells.

**Reboot Windows**
(0|Reboot Windows|MNU_WIN_REBOOT)
Shoot down Windows without confirmation and restart windows.

**Reboot computer**
(0|Reboot Computer|MNU_WIN_REBOOT 1)
Reboot your computer
(similar to Ctrl-Alt-Del).
The item **Reboot Computer** doesn't function with some Windows installations in the desirable way. Especially, if you have installed network driver (or similar), Windows prompts you to press Ctrl+Alt+Del to reboot the computer. There is no way to go around this.

**Quit - Exec - Win..**
(0| Quit - Exec - Win...|MNU_WIN_REBOOT 2)

You are prompt for a DOS command line, which will be executed after Windows is shoot down. After executing the DOS command line, Windows will be start again.

## Window-Manager

This Submenus are defined in the MNU-file ADDONS.MNU.

Submenu Programs

Arrange Icons, Tile Windows vert., Tile Windows hor. and Cascade Windows

Kill

Save Wins DT, AddDT and DelDT

Arrange Menus at Top and Arrange Menus at Left

Screensaver

## Programs

This Menuitems effects all opened and visible windows (programs)

With **Save All** you can ask all running programs for unsaved changes.
(0|Save All|MNU_CALLDLL USER SENDMESSAGE 0XFFFF 0X0011 0 0L)

With the menuitem **Minimize All** you can minimise all windows.
(0|Minimize All|MNU_DOWITHALLWIN 1)

With **Close All** you can close (terminate) all running programs.
(0|Close All|MNU_DOWITHALLWIN 2)

**Kill**
Please see: Kill

## Arrange Icons, Tile Windows vert., Tile Windows hor. and Cascade Windows

This menuitems functions like the buttons of the Task Manager.

```
(0|Arrange Icons|MNU_CALLDLL USER ARRANGEICONICWINDOWS @DESKTOPWINDOW
0|Tile Windows vert.|MNU_CALLDLL USER TILECHILDWINDOWS @DESKTOPWINDOW 0
0|Tile Windows hor.|MNU_CALLDLL USER TILECHILDWINDOWS @DESKTOPWINDOW 1
0|Cascade Windows|MNU_CALLDLL USER CASCADECHILDWINDOWS @DESKTOPWINDOW 0)
```

See also: Arrange Menus at Top and Arrange Menus at Left

## Kill

Be carefully with this menuitem! It really kills the windows; You aren't prompt for save-confirming! Maybe a killed application doesn't free all resources. If you press the Shift key while clicking onto the window. The window will be thrown out of the tasks list and the application has no change, to free its resources. It is recommended to reboot windows after a 'kill' action.
(0|Kill|MNU_MOUSE_CAP 1)

## Save Wins DT, AddDT and DelDT

With the menuitem **Save Wins DT** you can save all positions of the applications (windows) on the desktop. If you restart Windows, all Application will be restored exactly in the same positions (exceptionally of the applications are in [Autostart] or [Startup].
**Delete Wins DT**
Deletes all programs from the desktop list. After rebooting Windows, no program will be launched.
Alternatively you can press the **Shift**-key while selecting **Save Wins DT**.

This option has only one disadvantage. Of course (?!) the Menu can't figure out, which documents are loaded in the applications, the document-based applications will be restored empty.
Invisible windows (programs) will not be saved.
(0|Save Wins DT|MNU_SDT)

**Add Window to DT**
Save a single programs position to restore, when restarting the Menu.
After selecting this menu item, click onto the window, which should added to the Windows Desktop.
(0|AddDT|MNU_MOUSE_CAP 10)

**DelFromDT**
Delete a single window from restore-list.
After selecting this menu item, click onto the window, which should deleted from the Windows Desktop.
(0|DelDT|MNU_MOUSE_CAP 11)

## Arrange Menus at Top and Arrange Menus at Left

**Arrange Menus at Top**:
Minimises all pinned Menus and orders them on the top of the desktop.
If an pinned Menu is active, you can also use the **8** on the Numpad or the **F11**-key.

(0|Arrange Menus at Top|MNU_ARRANGE_TOP)

**Arrange Menus at Left**:
Minimises all pinned Menus and orders them at the left side of the desktop.
If an pinned Menu is active, you can also use the **4** of the Numpad or the **F9**-key.
(0|Arrage Menus at Left|MNU_ARRANGE_TOP)

# Virtual Desktop

(v1.3)

### Screensaver

This item only works, if you have installed a Windows screensaver in the desktop dialog of the control panel. If you select this item, the screensaver will be activated.
If you want activate another screensaver module, you just declare the module as commandline with following /s flag. E.g.:
C:\WINDOWS\SCRNSAVE.SCR /S

## Systemmenu

Also the System Menu is supported by using the right mouse button. The several actions (Restore, Move, Close etc.) are illustrated with icons. The menuitem "Switch to..." is expanded to the Tasks-Submenu. Besides this, you can stick the Window (hold on the top of the desktop).
If you activated the option **With hidden Tasks** in dialog **General Properties** in the section **Miscellaneous**, is the Menuitem **Hide** available to hide the window. You can unhide the window with Submenu **Tasks**.
In the dialog **General Properties** in the section **Menu - Hooks** you can declare a submenu, which should append at the system menu. (default "Systemmenu").
There you also can also appoint, if the system menu should also appear, if you click on the title bar of a windows with the right mouse button.
Unfortunately I haven't found a way, to get the current state (minimised, maximised or normal) of the window. This causes the menuitems of the system menu never showed deactivated. But the good news are, that if an application add own items on System Menu, they are also supported with the right-mousebutton-system menu.
If the application adds menuitems to its system menu, it will also be showed.
But if you use another desktop tool (maybe a similar one to New Menus for Windows), which adds also some items to system menu, this menuitems will normally not showed (it's because some tools are incompatible in this feature).

**Back**
If you select this item, the window will be put back at the last position of the window stack.

**Hide**
Hides the window.
You can unhide the window with the submenu **Tasks**.
This item is only available, if the option **with hidden Tasks** in the dialog **General Properties** is selected.

**Stick it** / **Unstick it**
Hold the window in the front of all windows.

**Fixed** / **Moveable**
Fixes the window in the visible **Virtual Desktop**, resp. makes it movable.
This item is only available, if the Virtual Desktop is used.

## Printer

In this Submenu all printers, which are installed, are listed. The active system printer is checked. If you select another printer, this printer will installed as the default system printer. If you press the **Shift**-key and select a printer, you can get the printer dialog. If the submenu 'Printers' is pinned, this has the side effect, that the Submenu will rolled up.

The Submenu Printer also supports Drag'n Drop with the File Manager: If you drag a file from the File Manager to an Printer, the file will be printed with this printer. If you want insert a new 'Printer'-Submenu, you have to declare MNU_PRINTERS as the file name of the Submenu.
(1|Printers|MNU_PRINTERS)

**Include a single Printer in a Submenu**
You can also include a single Printer a normal Submenu

1=0|My Printer|MNU_DEV Printerdevice
- 1=0| : A normal menuitem in a Submenu
- My Printer: Free description of the printer (Label of the Menuitem)
- MNU_DEV: internal keyword
- Druckerdevice: internal description of the printer.

The internal printer description of the active (default) printer you can find in the WIN.INI in the section [windows] after the device= statement.
Another possibility is highlighting a printer in the Submenu Printers and pressing the **F6**-key to get the internal printer description.

Samples:
1=0|The Fax|MNU_DEV FAXLINE,FAX,COM2:
2=0|Laser in File|MNU_DEV HP LASERJET III,HPPCL5MS,FILE:

For manually defined Printers item in a normal Submenu, you can also define an incon.

## History

In the Submenu "History" all commands (menuitems) you executed with the Menu are listed. In the dialog **General Properties** in the section **History** you can define the length of the Submenu History. If this value is zero, the history-feature will not used. The option **With internal commands** regulate, if also the internal commands should be listed. (This feature doesn't work properly).

If you press the keys Ctrl-h if the Menu is popped-up, you can jump to the Submenu History, If you press the keys Ctrl+Alt+h you can popup the Submenu History every time. (This are only defaults, which can be changed or deleted.)
(1|History|NIX)

If you call a program via an hotkey, this will not listed in the Submenu History.

## Control Panel

This Submenu lists the items in the control panel.

If the items doesn't react and you have Windows in an other language than English, you have to modify the correspondig Item in the RRKMENU.INI

F.e:
The orginal item is

1=0|Colours|CONTROL.EXE COLOURS<0;MAIN.CPL 0

If you have a German version, you have to modify this entry to:

1=0|Farben|CONTROL.EXE FARBEN<0;MAIN.CPL 0

Important is the second 'FARBEN', which is the German Label for the Colour-option in the Control Panel.

## Context-Sensitive Menus

This Submenus are defined in the MNU-file ADDONS.MNU.

See also: Associate a Window-Class with a Menu
Context-sensitive Add-Ons for New Menus for Windows

## Systemmenu of pinned Menus

This Submenu appears, if you hold down the right (or the middle) mousebutton over the title bar of a pinned Menu.
This Submenus are defined in the MNU-file ADDONS.MNU.

Arrange Menus at Top and Arrange Menus at Left

**Hide:**
With this menuitem you can hide the pinned Menu.
(0|Hide|MNU_CALLDLL USER ShowWindow @HWND 0)
(See also: Tasks)

**Stick it** and **Unstick it:**
With **Stick it** you can hold the Menu in the front of the desktop.
(0|Stick it|MNU_CALLDLL USER SetWindowPos @HWND -1 0 0 0 0 3)

With **Unstick it** a pinned Menu can be overlapped by other windows.
(0|Unstick it|MNU_CALLDLL USER SetWindowPos @HWND -2 0 0 0 0 3)

**Fixed** and **Moveable**
This menu item is only available, if the Virtual Desktop is enabled.
If you select the item **Fixed** the pinned menu always stays in the visible virtual desktop.

The menuitem **Close** do exactly what it says with the pinned Menu.
(0|Close|MNU_CALLDLL USER DestroyWindow @HWND)

See also: Stick Menus on the Desktop

## Editmenu of pinned Menus

You can get this menu, if you click with the right (res. middle) mousebutton on an item of a pinned Menu.

This Submenu is defined in the MNU-file ADDONS.MNU.

Edit Item
Insert Item
Add Item
Delete Item
Configuration

## Edit Item

If you want to edit (change) a Menuitem in a pinned Menu, you can click with the right (middle) mousebutton on this Menuitem. Select **Edit Item**.
In the Edit dialog you can change the properties of the Menuitem.
(0|Edit Item|MNU_EDIT)

See also: Edit an Existing Menuitem

## Insert Item

If you want to insert a new Menuitem in a pinned Menu, you can click with the right (middle) mousebutton on the Menuitem, which marks the point to insert. Select **Insert Item**.
In the Edit dialog you can declare the properties of the new Menuitem.
(0|Insert Item|MNU_INS)

See also: Insert a New Menuitem

## Add Item

If you want to add a new Menuitem in a pinned Menu, you can click with the right (middle) mousebutton on the Menuitem, which marks the point to add. Select **Add Item**.
In the Edit dialog you can declare the properties of the new Menuitem.
(0|Add Item|MNU_ADD)

See also: Add a New Menuitem

## Delete Item

If you want to delete a Menuitem in a pinned Menu, you can click with the right (middle) mousebutton on the Menuitem to delete. Select **Delete Item**, an confirm the dialog to delete.
(0|Delete Item|MNU_DEL)

See also: Delete a Menuitem

## Configuration

Pop up the dialog Configuration.
(0|Configuration|MNU_CONFIG)

## Startup = submenu Autostart

If you run the Menu as windows shell, all items in the submenus, which calls **Autostart** or **Startup** are executed at the beginning. These submenus has to be defined in the RRKMENU.INI.
See: Starting Options

## Keyboard

The keys to configure the Menu only work if you have popped up the Menu and highlighted a Menuitem - respectively a pinned Menu has the focus.
If you haven't a mouse look at Using the Menu without a Mouse.

**F1** shows the Help
**F2** invokes a dialog to change the General Properties

**F5** brings up a dialog to Configure the Menu and edit a Menuitem
**F6** Edit a Menuitem
**F7** Add a Menuitem

**Del** Delete a Menuitem

**Ins** or **F4** Insert a Menuitem

**F9** or **4 on the Numpad**:
Arrange all pinned Menus left on the Desktop
**F11** or **8 on the Numpad**:
Arrange all pinned Menus left on the Desktop
**F12** Refresh a pinned Menu.

**Alt + Pause** to popup the Menu by keyboard

**Ctrl and Shift**

- while selecting a Menuitem:
  Hold down the Ctrl-key to change the command line before execution
- If don't use the Menu as Windows-Shell, you can also quit Windows with the Menuitem **End**. If you push the **Ctrl**-key while selecting this menuitem, Windows will shoot down. If you push the **Ctrl-** and the **Shift**-key, Windows will shoot down without a confirmation.
- If you open the Submenu **Tasks** while holding the **Shift**-key down, also the hidden windows will list in the Submenu.
  If you click on a Menuitem of the Submenu Tasks and press the **Ctrl**-key, the corresponding program will be closed. If you click on an item and press the **Shift**-key the program (window) will be hidden.
- If you click on a Menuitem of a pinned Menu with the **Shift**-key, the menu will roll up after executing the Menuitems command line.
- If the Menu starts and you press **Shift**-key, Wins DT (saved with **Save Wins DT** or the internal command MNU_SDT) will not be restored. Also - if the Menu runs as Shell - the items in Autostart - will not execute.
- If you execute a Menuitem of the Submenu Printer and you hold down the **Shift**-key, you can get the to printer setup dialog.
- To provide a better compatibility with other desktop tools like PC Tools for Windows or Norton Desktop, you can disable the popup with the right mouse button, if you press the **Shift**-key. If you press the shift-key you can access the original popup menus of other desktop tools.
- To refresh the Program Manager groups, press the **Shift**-key while opening the submenu 'Progman'.
  If you want to refresh a Program Manager Group, use the same method.
  If you enabled the window activation via **XMouse**, you can temporary deactivate this feature with holding down the **Shift**-key.
- **Virtual Desktop**
  Click with the **right mouse button** onto a virtual desktop to switch to it.
  Doubleclick with the left mouse button to show or hide the title bar of the virtual desktop.

# The Dialog Menu - General Properties

**How to get the Dialog:**

- Press the **F2**-key in the popped up menu.
- Press **Gen. Props** in the Configuration Dialog (Accessible through **F5**).

**Explanations**

In this dialog you can change the General Properties of Menu.

The General Properties are divided in following sections:

Style
Menu Fonts
Menu Hooks
Mouse
Miscellaneous
History
File Viewer
Virtual Desktop

## Style

Graphic
Icons
Icon Size
Style

## Graphic

The menu has a 3d-effect.
With Fonts you can choose a Font for the menu items.

Comment: The pinned Menus are always shown with the 3d-effect.

## Icons

One of the highlights of the Menu is the capability to automatically show the Icons of Programs and Documents in the Menu.
But if you have a slow computer and/or use disk compression software, you may not want this option.
Look also at The meaning of the Icons

**Icon Size**

The numbers correspond to the size of the icons.
The normal size of the Icons - as shown by the Program Manager - is 32 pixels. 24 and 16 pixels are smaller.

## Fonts

A standard dialog for choosing the menu font.
The standard font is MS Sans Serif 8 pt.

## Style

**Standard** is a simple style, but it has the advantage that the menu items are small and the submenus are built quickly.
The style **OpenLook** is overtaken from a UNIX X-Windows GUI.
The style called **Beta** is just a one-evening-production.
The style named **Motif** has some similarities to the X-Window GUI with the same name.
If you change the style to or from **Standard** all icon database files RRKIDX.* and RRKDAT.* are deleted.

## Menu Fonts

If you selected a graphical style, you can customize the menu font.

In the right field - **Menu** - you can define the fonts of the menu items and the alignment of the label in the menu.

In the left field - **Title** - you can select the font of the title of a submenu and the alignment of the title label.
If you don't want a label in the title of a submenu, deactivate the option **with title label**.
With the option **fit to title label** you can select, if the width of a submenu should be a least the width of the title label.

**Comment:**
The alignment of the menu font is not supported if you select the style **Beta**.

By default the font is "MS Sans Serif" in the items and "MS Serif" in the title label.

See also:
Style
Colors

## Colors

In this dialog you can customize the colors of the menus. If you push one of the buttons you get a standard Windows dialog to select a color.

**Menu**
Here, you can choose the background color for the menus. By default this is gray.

**Highlighted Menu**
If you highlight a menu with the mouse, the item will be shown in a different manner depending on the chosen menu style. You can select the background color of a highlighted menu with this button.
If you use the style **Motif** this option has no effect.

**Menu Text**
The text color of a normal menu item. By default this color is black.

**Highlighted Text**
To have a better contrast to the background of an highlighted menu item you can select a different font color for highlighted menu items.

See also:
Style
Menu Fonts

## Menu Hooks

In this dialog you can configure the behavior of the mouse in when used with other windows applications. You can also define your root- or main menus, which should appear if you press a mouse button.

HotKeys
Menu Hook (Drop Down Menus)
Context Sensitive
System-Menu also pops up over title bar
Standard menus

## Hotkeys

The Menu can be used without a mouse (on notebooks, maybe). Using this option, you can popup the Menu by pressing the Alt and the Pause key. To navigate in the Menu and submenus, use the cursor keys on the keyboard.

See also: Using the Menu without a Mouse

If you want to use Hotkeys to launch programs this option must be activated.

### Menu Hook (Drop Down Menus)

If this option is on, you can select a drop-down menu of a normal Windows program with the right mouse button. This Menu has a pin to stick it on the desktop.
See also: Stick normal Drop-Down-Menus with a Pin.

**Context Sensitive**

If this option is on, the context sensitive popped up menus, described in the file ADDONS.WRI, are enabled.

See also: Using Context-Sensitive Menus.

Context-sensitive Add-Ons for New Menus for Windows

**System-Menu also pops up over title bar**

If this option is activated and you click with the right mouse button on a windows title bar, you get the system menu. Otherwise you get system menu only if you click with the right mouse button on the system menu field.

## Standard Menus

There are three types of standard menus:
The **root**- or **main menu**, which appears if you press the right or middle mouse button.

If you use the **middle mouse button** (or an emulation of the middle mouse button) to obtain the **main menu**, you can define a second menu, which will be popped up with the **right mouse button**. Unlike the main menu, the **submenu for the right mouse button** only appears, if the mouse cursor is over the the desktop.

If you open the system menu of a window with the right mouse button, you get an advanced system menu. You can define your own menu items, using **Appending for the System-Menu**. These items are defined in a submenu.

If you want to change one of this standard submenus, place the cursor in the corresponding text field and select **Browse** to pick another submenu.

## Mouse

In this dialog you can select different switches for using the mouse.

**Menu - Mouse button**
Release Mouse after Popup
Middle Mouse button

**Emulation of the middle mouse button**

**XMouse window selection**

**Popup-align**

### Release Mouse after Popup

If you check the option **Release Mouse after Popup** the right mouse button is released after you popup the Menu with the right mouse button. Otherwise you can select a menu item by holding down the right mouse button.

### Middle Mouse button

If you have a mouse supporting 3 mouse buttons, it is recommended to use this option. To pop up the menu, you can use the middle mouse button instead of the right mouse button.
See also: Avoid conflicts with other Programs.

## Emulation of the middle mouse button

You can emulate the middle mouse button with:
Ctrl + right mouse button
or
left + right mouse button

The order of the Ctrl / left + and right mouse button is important: First press the Ctrl-key / the left mouse button and then the right mouse button.
This option only works if the option **middle Mouse button** is activated.

## XMouse window selection

If you want work with many programs at the same time and you often have to switch between these programs, it can be annoying to have to click with the mouse to select the desired window.
In UNIXs X-Window system, the mouse can activate windows only by moving the cursor over the window.

Two mode are available:
**pop up to foreground**
The window under the mouse cursor will be activated and will pop up in the foreground of the window stack.

**only activate**
The window will get the focus, but it will not come into the front. This means that, if an activated window is under another window, after activating the window, it will still be under the other window. If you want to bring the window to the front, youll have to click with the left mouse button over the windows title bar.

**Note**
These features break with the styling guides for Windows. Some programs may not work properly if you use this feature.
**You can temporarily deactivate the XMouse window activation by holding down the shift key.**

### Popup-align

If you pop up the menu with the right or middle mouse button, the menu appears aligned to the mouse cursor. With this buttons you can select if the menu should appear **left, right or middle** to the mouse cursor.

## Miscellaneous

Index in RAM
Real Move
Save DT. (Desktop)
As Shell
Confirm End
Dynamic PMGs

With hidden Tasks
Task only with main windows

**Index in RAM**
If you have enough RAM, it is recommended to use this option: The menu will be faster this way.

### Real move

If you have a fast graphics adapter in your computer, you can use this option to simulate a real-time movement of the pinned menus as in the NeXT Systems.

**Save DT. (Desktop)**
Saves the desktop layout when leaving the menu, so it can be restored latter when restarting the program.
If you want to save one particular layout of the sticked menus, you can arrange them, quit the menu, restart it and then uncheck this option. From then on, the menu starts every time with the same layout.

### As Shell

To use the menu as shell instead of the Program Manager.
See also: Starting Options

### Confirm End

If you use the Menu as Shell and if this option is activated, the Menu will prompt you for confirmation before quitting Windows.

### Dynamic PMGs

If you have a fast computer or you don't use the icons in the menus, you can activate this option. This has the advantage that the submenus "Progman" and the Program Manager Group Submenus are always up to date because they are dynamically read out of the group files (*.grp). I don't recommend the use of this option, though.

See also: The Program Manager as Menu.

### With hidden Tasks

Enabling this option allows the hidden tasks (windows) to be (also) listed in the submenu Tasks.

### Task only with main windows

If you have activated this option, only the main windows of the applications will appear in the task list in the submenu item **Tasks**.

## History

**With internal command**
If this option is activated, the internal commands will also be recorded in the submenu History

**Length of the Menu History**
You can specify the maximum length for the submenu History.
If the length is 0, no executed items will be recorded.

## FILE VIEWER

You can specify a File Viewer and a File Editor.
If you choose a file which is not a program or a document, the file is opened with the File Viewer by default. If you press **Shift** while selecting it, the file will be opened using the File Editor.
The File Viewer and File Editor are also used in the context sensitive menus for the File Manager.

## Virtual Desktop

See also: **Virtual Desktop**

**columns / rows**
The numbers of columns and rows. You can have a maximum of 8 rows by 8 columns.

**Switch to active window**
If you activate a window - using the submenu task; with the keys Alt+Tab or by double-clicking into a window in the virtual screen - and the activated window is not in the current desktop, the virtual desktop will automatically switch to the appropriate screen - the one containing the activated window.
If you don't want this to happen, uncheck this option.

**Always fixed icons**
If this option is on, all minimized windows will moved into the current virtual desktop.

**OK**
To confirm the changes in this dialog.

**Cancel**
Selecting Cancel will undo all changes.

# The Dialog - Delete Menuitem

## How to get the Dialog:

- First, press the **F5**-Key in the popped-up Menu,.and then select **Delete**;

*or*:

- Press the **F4**- or the **Del**-Key in the popped-up Menu;

*or*:

- In a pinned Menus select the corresponding Menuitems of the Editmenu.

## Explanations

This Dialog confirms the deleting of a menuitem.

## (Sub-)Menu

The submenu, which contains the actual selected Menuitem, is shown here.

## Label

The Label of the actual selected Menuitem is placed here.

## Filename

The command line of the actual selected Menuitem is placed here.

## Working Directory

The Working Directory of the actual selected Menuitem is placed here.

# The Dialog - Delete a Submenu

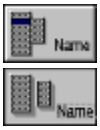
## How to get the Dialog:

- If you have selected a popped-up Menuitem, which contains a Submenu and you press the Del-Key, the Menu asks you, if you want to only delete the Menuitem, without deleting the whole Submenu (**Name**), or if you also want to delete the entries of the selected Submenu (**All**).

## Comment

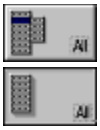
It is recommended to delete the whole Submenu if you don't need the Submenu at another place in your Submenu structure. Remember, the RRKMENU.INI must not become bigger than 32 KB.

## Name




By confirming with **Name**, only the Menuitem, which links to the (Sub-)Submenu in the actual (Sub-)Menu, will be deleted. If you Insert the (Sub-)Submenu in another (Sub)Menu of your Submenu structure, you can access the old Entries. This might be a way to move whole Submenus to other places in the menu structure.

## All




By confirming with **All**, the whole Submenu will be deleted. This will have the consequence, that any other Menuitems with a link to this Submenu will become empty.

# The Dialog Menu - Edit Menuitem

**How to get the Dialog:**
There is more than one way:

- At first select, in the popped-up Menu, a Menuitem, then press the **F5**-Key and choose (in the dialog Configuration) one of the buttons: **Add**, **Insert** or **Edit**.

*or*:

- At first select, in the popped-up Menu, a Menuitem, then press the **F6**-Key to **edit (change)** a Menuitem.
- ... the **F7**-Key to **add** a new Menuitem
- ... the **F4**- or the **Ins**-Key to **insert** a new Menuitem**.**

*or:*

- Choose, in a pinned menu, the corresponding Menuitem of the local Editmenu.

**Explanations**
In this Dialog you can insert, add or change a Menuitem.

**Comments**
By using the "Drag and Drop-method" from the File-Manager, you can also add or insert new Menuitems into a pinned-Menu, without using this dialog.
See: Drag'n Drop: Insert files in a Submenu

SubMenu
MenuFile
Label
Filename
WorkingDirectory
Hotkeys
Icons
MenuType
Properties

**(Sub-) Menu**
The Submenu, which contains the current selected Menuitem, is shown here.

## Menufile

Submenus can also be defined in files other than the RRKMENU.INI. To do this, you should declare the **Filename** of the Menu-file, which contains the Submenu to be used within a Submenu for .
The Menufiles must have the extension *.MNU.
If you edit a Menuitem, which is defined in a MNU-file, the MNU-file will be shown in this field.

**Label**

The Label is shown in the Menuitem of the Menu. The Label must not be longer than 30 chars. Internally, (As with a link to a Submenu) the case makes no difference.
If there is the char "&" in the string of the label, in the menu the following char will be showed underlined. The label "My &Programs" will be showed as "My Programs". If you want to see the char "&" in the menu item, you have to write two "&"'s. The label "files && documents" will be showed as "files & documents".

**Filename**
Normally a program-file or a Document- is executed by choosing a Menuitem.
(NIX is a short form of 'Nothing').
("MNU_"... is reserved for internal command)

### Working Directory

You can declare a standard working path for programs.
(This will only have an effect for standard **Menuitem**s.)

## Hotkeys

A Hotkey is a combination of keys, that execute a command line, its inclusive parameters, working directories and window sizes - every time the Hotkey is pressed.

To define a Hotkey, put the cursor in the Hotkey edit-box and press the key combination desired.

If the actual Menuitem is a Submenu, you can use this edit-box to declare a ShortCut to the Submenu. ShortCuts can only be used if the menu is still popped-up.
To define a ShortCut for the Submenu, press the **Ctrl**-key and a char. If this ShortKey is already associated with an other Submenu, the old association will be overwritten.
To define a Hotkey for the Submenu you must check the option **Global**.

See also: Using ShortCuts or Dialog ShortCuts.
See also: Using Hotkeys or The Dialog Hotkeys.

**Icon...**
Opens the dialog Icons to assign an Icon to the current Menuitem.

See also: Declare Icons for Non-Windows Programs.

**Menutype**
Normal Menuitem
Submenu
Horizontal Dividing Line
Vertical Dividing Line
Searching Directory
Drive Directory
Program-Manager
Program-Group

### Normal Menuitem

A normal Menuitem needs a **Label** and a **Filename**.
The **Label** depends on your imagination. (f.e.: `My fine Program`)
Use **Browse** for picking a Program- or a Document- File for **Filename**. When executing other Files, than Programs or Documents, the execution is preceded with a Beep.
You can also declare Arguments after a whitespace in **Filename**: f.e.: `c:\mydir\myprogr.exe /v /r myfile.dat`
In the field **Working Directory** you can declare a path for the program.
Also you can choose the window size of the starting program in **Window-Size**.

See also: Internal Command

## Submenu

A Submenu is a cascading menu.
By choosing a Menuitem that contains a link to a Submenu, the Submenus are defined by their **Labels**. f.e.: `Games and More`
Internally there are no differences between capital and small letters (the case). A declaration for a **Filename** is not necessary (NIX is 'short' form for "nothing").
But if you want to define the submenu in an external Menu you have to declare the MNU-file as **Filename**.

With Submenus, instead of changing **Window Size** you can change the **Style of a Submenu**.

### horizontal Dividing-Line

Use this, if you want to structure your Menuitems in several horizontal sections.
Note, though, you must only use the suggested entries in **Label** and **Filename**.

### Searching Directory

This option generates Menuitems by using DOS ambiguous file-names.
If you type `c:\mydir\*.txt` in the field **Filename**, then in its Submenu all matching textfiles are listed in the order, you defined with the Properties of the Menuitem. Entries in the field **Label** won't have an effect, because they are replaced by the found filenames (listed without the path).
You can use **Browse** to select the directory and a file.
(Comment: You can collect your actual project files in a Submenu, stick this Submenu on the desktop, and, thus, you don't need to use the Menu 'File - Open' in an application.)

## Drive Directory

This function puts a little 'File-Manager' in a submenu. In this submenu all directories are shown first, and executable files and documents are listed last in a alphabetical order.
In the field **Filename** you have to type the starting directories for example `c:\` or `c:\mydir\mysubdir`. (After selecting this option, you can use **Browse** to choose the starting path by selecting any Dummy - file.)
In the field **Label** you can use the defaults offered by **Browse** or you can declare your own Label.

### Window - Size

Window-Size can only be chaged on standard **Menuitems** (i.e. only on executables, or documents assigned to executables, not on submenus, etc.).
If the Menuitem is a Submenu you can declare the **Style of a Submenu** at this position.

### vertical Dividing Line

You can use this item to separate the Menuitems below the vertical Dividing Line in a new column.
(This feature is still 'Alpha')
Note, though, you must only use the suggested entries in **Label** and **Filename**.

## Progr.- Manager

You can include the Program-Manager in the menu with this option.
You must use the suggested default in the field **Filename**.
You can use the default in the field **Label**, or you can change it.
Please refer to The Program-Manager as Menu for more detailed information.

**Progr.- Group**

You can include a single Program-Group from the Program-Manager in the Menu.
Use **Browse** search for an existing '*.grp'-file in your Windows-Directory.
In the field **Label** you can use the default, offered by **Browse,** or you can declare your own Label.

Please refer to A single Program-Group as Menu for more detailed information.

**actual Icon**
The actual Icon of the menuitem is shown here.
(Sometimes, when you change an Item, the icon of the old item will still be shown)

## Properties

If the Menuitem is a Normal Menuitem or a Searching Directory then for each property selection:
**Normal**
The program will start in its normal size.

**Minimized**
The program will start minimized

**Maximized**
The program will start maximized

**Hidden**
The program will start hidden. Use the Submenu Tasks to switch to a hidden program.

If the Menuitem is a Submenu or a a Drive Directory: then for each property selection:
**Normal**
The Submenu and the Menuitems are showed in the default style.

**o.Icons**
The Menuitems of the Submenu will be showed only with the icons.

**o.Text**
The Menuitems of the Submenu will be showed only with the describing text.

**NoPin**
The Submenu will not have a pin to stick it on the desktop.

**Locked**
You can't edit the Menuitems of the Submenu with dialogs.

**All Files**
Is only available if the Menutype is Drive Directory
All files are listed even if they are neither programs or documents. Also hidden files or system files will be showed.

**by name**
All found files are listed by name.
This option is only available, if the type of the item is drive path or searching directories.

**by type**
All found files are listed by the type of the file (file extensions).
This option is only available, if the type of the item is drive path or searching directories.

**by time**
All found files are listed by time (last modified) of the file.
This option is only available, if the type of the item is drive path or searching directories.

**by size**
All found files are listed by size of the file.
This option is only available, if the type of the item is drive path or searching directories.

**reverse**
All found files are listed by name, type, date or size in reverse order.
This option is only available, if the type of the item is drive path or searching directories.

**OK**
After editing the Menuitem, save any changes by clicking on the OK button.

# The Dialog Menu - Execution

### How to get the Dialog:

You must press the **Ctrl**-Key while executing a Menuitem in order to get the execution dialog.

### Explanations

You can change the filename and its parameters in the dialog before executing it.
It is also a great replacement for the Program-Manager "File -Run" command (or similar, I do not have the English Version of Windows)

### Command line

Place the Filename of the program or the document here.

### Parameter

Place whatever flags or filename expected by the program.

### Working Directory

Some programs needs to start in a special directory or they can't find their files.

### OK

After confirming, the program executes, and the assigned-document, if any specified, will also be open.

# The Dialog Menu - Configuration

### How to get the Dialog:

- In the submenu **Menu Setup** select the item **Configuration**

*or:*

- Press the **F5** Key in the popped-up Menu.

*or:*

- Select the item **Configuration** in the Edit-Menu of pinned Menus.

*or:*

- Select the item **Configuration** in the system menu of the program icon.

### Explanation

With this dialog you can edit the menu structure. You also have access to some other configuration options of New Menus f. Windows.

In the field **Menu file**, you can indicate the file where your menus are defined. By default the RRKMENU.INI file is shown. If you want to choose a different one, click on the little yellow folder button. The next step is to choose the starting point of the menu structure, you wish to edit. By default this is the "Main menu" submenu. Open the drop down list to select another submenu defined in the menu file youve selected.

But you can also edit submenus of the menu defined as starting point of your menu structure.

If you double-click on an item in the list representing a submenu, this submenu will be opened. If you then highlight a menu item you have following options to edit it:

### Copy

The highlighted menu item will be copied to an internal clipboard.
Choose another point in the menu structure and press **Insert** or **Add**, to place the item from the clipboard.

### Cut

Works in the same manner as **Copy**, with the difference that, the item which is copied into the clipboard is also deleted.

### Up

The highlighted menu item will be moved up. This is valid only for the current submenu.

### Down

Very much like **Up**.

### Edit Menu item

Opens a dialog to edit the current menu item.
Note: If there are two identical menu items in one submenu, and you edit one of them, changes will be reflected on the first of the two.
See: Edit a menuitem
Dialog Menu - Edit Menuitem

### Delete Menuitem

Deletes the actual Menuitem.
See: Delete a Menuitem
Dialog Menu - Delete a Menuitem

### Add Menuitem
Opens a dialog, to add a new Menuitem below the position of the current Menuitem.
If youve copied or cut a menu item with **Copy** or **Cut**, this menu item will be added.

See: Add a menu item
Dialog Menu - Edit a Menu item

### Insert Menuitem
Opens a dialog to insert a new menu item, above the position of the current menu item.
If youve copied or cut a menu item with **Copy** or **Cut**, this menu item will be inserted.
If you want to **insert a menu item on an empty submenu**, you have to highlight the submenu item in the parent submenu. After clicking on **insert** and filling out the edit dialog you are asked if you want the new item as the first item of the empty submenu.
See: Insert a Menu item
Dialog Menu - Edit a Menu item

---

### General Properties
Opens a dialog where general properties of the Menu can be changed.
See: Dialog General Properties

### Associate...
Calls a dialog that allows you to associate documents (declared by the file extensions) with applications.
See also: Dialog - Document - Association

### ShortCuts

Opens the dialog ShortCuts.

### Hotkeys
Opens the dialog Hotkeys.

**Browse:**

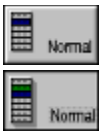

This button will open a standard dialog for picking a file.

# The Dialog - New Menuitem in an empty Submenu

### How to get the Dialog:
If you insert a new Menuitem, and you highlighted an empty Submenu, this dialog appears.

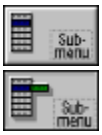
### Explanations
To insert the first Menuitem in an empty Submenu, you have to highlight the empty Submenu and press the Ins-key.
After declaring the new Menuitem in the Edit dialog you can choose, if you want this new Menuitem as the first Menuitem in the empty Submenu.

### Normal




The blue Menuitem contains the empty Submenu. If you choose **Normal**, the new Menuitem (green) will be placed at the above position. The empty Submenu will still be empty.

### Submenu




The blue Menuitem contains the empty Submenu. If you choose **Submenu**, the new Menuitem (green) will become the first Menuitem of the empty Submenu.

# The Dialog - Document Association

### How to get the Dialog:

- Press the **F5**-key in the popped-up Menu.
- Select **Configuration** in the Systemmenu of the program icon (a black Arrow).

*or:*

- Select the Menuitem **Configuration** in the Submenu **Menu Setup**.

*or:*

- Press the **F5**-key and select the **Configuration** dialog in the popped-up.

*or:*

- Click on a pinned Menu with the right (middle) Mousebutton and select **Configuration** in the local Editmenu**.**
- Select **Associate**

### Explanations

In this dialog you can declare your Windows Documents by associating a file extension with a program.

### Comments

Documents are defined by their file extension. For example: A file with the file extension `.wri` is a Write - Document associated to the program WRITE.EXE.

## How To:

### Define a Document

- Declare a File Extension in the field **File Ext.**
- Select a Program in the list of **Associated Programs**

*or:*

- Use **Browse**, to pick up a program file.

### Undefine a Document

Select, in the left list, a File Extension to detach and click on the button **detach**.

## The Elements of the Dialog

### File Ext. (File Extension)

In this field You must insert the File-Extension.

### List of the existing File Extensions

All File Extensions that associate documents and programs, are listed here.
If you click on one File Extension, in the list **Associated Programs** the Program associated with the document will be highlighted.

### Associated Programs

In this field you must insert the Program filename.

### List of the associated Programs

Here all Programs listed, that are registered in REG.DAT

### Associate


If You have declared a File Extension in the field **File Ext.** and a program in **Associated Programs** You

can define (associate) a Document.

**Detach**


If You have declared a File Extension in the field **File Ext.** and a program in **Associated Programs** You can undefine (detach) a Document.

**OK**

To close the dialog.

Querverweis von VKN_DIALOG_HELP_ID auf vkndlg

## What you have to read / know

Chapters you may have to read, to use the Menu:

- You can get an overview of the features in: The features of the Menu.
- In the chapters Quick Tips, the essential functions are described.
- The chapters Bugs, Limits, Help may protect you from unwelcome surprises.
- The Chapter Avoid Conflicts with other Programs may give you hints to do exactly that.
- You have to know the Edit keys for customizing your menu.
- In the chapters How to do the important Actions are described step by step.

# Introduction to New Menus for Windows

Please read the description in the file README.WRI

Windows at your fingertips: New Menus for Windows 1.28 is a Windows Shell, similar to some window managers of X-Windows (GUI of UNIX e.g. Motif, OpenLook).

**The Menuing System**
The kernel of the program is a replacement for the Progman, in the form of a popup menu, which you can popup at any time with the right or middle mouse buttons. This popup menu can have as many submenus (like subdirectories in a filesystem) as you like. The menu items can be programs, documents or function calls. The popup menu is not a simple, normal windows popup menu. It doesn't look like one and its behavior deviates. An icon and description are displayed for all programs and documents. Ad libitum these icons can have the size of 32, 24 or 16 pixels - it is even possible to represent menu items as animated icons. By default the icons of the programs or its assigned documents are shown. You can customize these icons. The font and font size is also configurable. For each submenu, you can choose if you want to see only the text, the icon or both. You can choose the look of your popup among 4 styles. One of these simulates the design of Motif and another is similar to OpenLook.
A nice feature of OpenLook is also implemented in NMFW: Every (sub-)menu can be stuck onto the desktop. You can install new Menu items in much the same way you would in Program Manager (e.g. Menu items which launch new programs). Installation of new menu items is supported by dialogs or via drag'n drop. The first time you run NMFW, the groups of the Program Manager are converted to menu items. To work with documents, you dont have to install each file as a menu item - New Menus for Windows supports wild cards, which automatically fill submenus with matching files. It is also possible to include parts of a filesystem into the menu structure: The subdirectories are represented by submenus and their files appear as the submenus items. These dynamically built menu items can also be sorted by name, extension, date or size (also in reverse order). If you stick a (sub-)menu onto your desktop you can use it as a buttonbar, which also supports drag'n drop with the File Manager. With a menu item you can do more than just launch a program or open a file: you can declare Keyboard macros, call Windows system functions or functions from any DLL. In the Tasks-Submenu you can switch between your running applications - also shown with their icon and title - hide windows, or terminate the applications. In another Submenu you can switch between your installed printers, call its property-dialog or drag files to print on a specified printer. In a submenu called History, all executed menu items are listed as a command history. Other special features are only listed by name: Rebooting Windows, rebooting DOS; quit Windows - execute DOS-program - restart Windows, Save all documents in all programs; Close all applications; the layout functions of Taskman; Save Desktop (window positions); Kill a window; etc.
The access to the Submenus is very customizable, several mouse button / keyboard combinations are supported (also an emulation of the middle mouse button). You can assign a global hotkey to each submenu or menu item (in combination with keyboard macros you can also redefine your keyboard layout). To jump quickly from one Submenu to another, you can use shortcuts.

**Enhancement of the standard Windows applications drop down menus**
If you select a normal drop down menu of a windows application with right mouse button (e.g. "Edit" of Write) you get a new designed menu. Besides the nicer (customizable) look of this menu you can stick it onto the desktop and use it as an sort of button bar (e.g. if you have to switch often between several documents in a multi-document based program, you can stick the menu "Window" onto the desktop to have quick access). The system menu is improved - the standard functions are illustrated with icons, instead of the item "switch to..." you get a Submenu with all running Tasks, you can hold windows to stay on top (like the clock) and you can append more menu items and submenus to the system menu.

**Enhancement of windows controls with context sensitive popup menus**
More and more applications use the right mouse button to give a quicker context sensitive access to the objects the mouse is pointing at. New Menus provides some context sensitive menus which have been 'forgotten' by Windows:

Edit controls (e.g. notepad): Copy; Cut; Insert; Undo; Mark all; Search; Save (ANSI, ASCII, UNIX); convert to lower or upper case; statistics.

List boxes: Copy the string of the marked item.

Scroll Bars: Up, down, left, right.

File-Manager: Copy marked files / directories as text string (which
you can use as arguments to other programs); Load file into the File Viewer
or File Editor; Print File; file operations (copy, move, etc.).

DOS-Box: Mark; Copy; Insert

It is relatively simple to customize your applications with other context sensitive menus.

**Multi-User Configuration**
New Menus for Windows supports different configurations for different users and network administration. You can supply individual INI-files in the command line (also as an argument for the WIN.COM). Each user can modify its own configuration, but a system administrator can also maintain predefined menu files, in a directory in the path, which can be linked as submenus to individual menu structures. These external menus can be locked against changes. If a new application is installed in the network, the system administration will only have to modify one file.

**Other features**
New Menus has very lean runtime resource requirements. It needs at minimum ca. 40kB non discardable RAM and 2% of the system resources, which doesn't depend to the size of the menu structure. You need Windows 3.1(1) or WfW. New Menus supports all mice and trackballs, but you can also use the Menu without a mouse.

## Howtodo - Index

## Insert a new menu item

**In standard, non-pinned menus:**
Comment

- Select the menu item menu Setup / Configuration, where you can edit the menu structure. Choose the position, where to insert a new item and select **Insert**.

Another way:

- At first you have to popup the menu with the right (or middle) Mousebutton.
- After that, you have to select (highlight) a menu item with the left Mousebutton to reference the point of insert.
- At the same time, push the **Ins-Key**.
- and release the Mousebutton.

- In the following dialog you can declare the characteristics of the new menu item.

**In pinned menus:**
Comment
*Either:*

- You can highlight the menu item.
- After released the Mousebutton, push the **Ins-Key**.

*Or:*

- Push over the menu item to insert the right (or middle) Mousebutton,
- and select in the Editmenu **Insert Item**.
- In the following dialog you can declare the characteristics of the new menu item.

*Or:*

Use the Drag'n Drop Method with the File-Manager.

If you want insert a menu item in an empty submenu, you have to highlight the submenu entry or the title bar of the empty submenu. If the submenu is an external menu (stored in a *.MNU file) you have to highlight the title bar of the submenu.

## Add a new menu item

There are the same methods, described in Insert a new menu item, with following differences:

- Select the menu item menu Setup / Configuration, where you can edit the menu structure. Choose the position, where to insert a new item and select **Add**.

Another way:

- You have to press the **F7**-key instead of the Ins-key.
- The new menu item will be add **after** the current highlighted menu item.

## Edit an existing menu item

There are the same methods, described in Insert a new menu item, with following differences:

- Select the menu item menu Setup / Configuration, where you can edit the menu structure. Choose the position, where to insert a new item and select **Insert**.

Another way:

- You have to press the **F6**-key instead of the Ins-key to edit the current highlighted menuitem.

## Delete an existing menu item

**In standard, non-pinned menus:**

Comment

Comment to delete a Dividing Line

- Select the menu item Menu Setup / Configuration, where you can edit the menu structure. Choose the position, where to insert a new item and select **Insert**.

Another way:

- At first you have to popup the menu with the right (or middle) Mousebutton.
- After that, you have to select (highlight) the menu item to delete.
- At the same time, push the **Del-Key**.
- and release the Mousebutton.

- Confirm following Dialog.

**In pinned menus:**

Comment

*Either:*

- You can highlight the menu item
- After released the Mousebutton, push the **Del-Key**.

*Or:*

- Push the right (or middle) Mousebutton over the menu item to delete,
- and select in the Editmenu **Delete Item**.
- Confirm following Dialog.

**Comments**

- If you can't remember all the Edit key, you also can choose the actions via **F5** and the dialog Configuration.
- Naturally you can't edit menu items in Drive Directory.
- If you want to insert a new menu item in an empty submenu, highlight the menu item, which contains the empty submenu and press **Ins**.
- If you insert, add, change or delete a menu item in a Progman-Group, the changes will also effect the Program-Manager.

**Comments**

- Searching Directories and Drive Directories can't be edit in pinned menus.

### Comment to Delete a Dividing Line

It is a little bit difficult to select a **horizontal Dividing Line**, but principle possible. The selection of a horizontal Dividing Line has no visual feedback by highlighted the menu item.
To delete a **vertical Dividing Line**, you have to select the continuation of the Head of the menu in the second column.

## Drag'n Drop:
## Open any File using the File-Manager and a Pinned menu

Often I want to view the contains of a file, that is not a document. If the file is not to huge, I start the notepad and use File - Open. If the file is bigger than 32 KB I have to use an other Editor. This seems me to be to complicated. Why not use a simple Drag and Drop with the File Manager, to open any file with any Editor?. You can create a submenu (maybe called 'Browsers') and install all your preferred Text- Hex- and docs-viewer in this submenu. You can use pinned (Sub-)menus as Drag'n Drop panel. If you drag files from the File Manager and drop them over an menu item, the files are send as arguments to the program installed as menu item. Of course, You can also install your Zip- and Unzip batch files in one of these submenus.

- Open the submenu, which contains a menu item with a file viewer (f.e. NOTEPAD.EXE) and stick it on the desktop by using the pin.
- It is recommended to click a second time on the pin, so it is complete pinned. The pinned menu is now every time in the front of the desktop.
- Open the File-Manager.
- Choose the file to open in the File-Manager and
- Drag the file with the left Mousebutton over the menuitem, which contains the file viewer and drop the file.
- Now the program starts and normally opens the file.

Of course the file viewer must handle the file. Try to open a text file with Paintbrush will fail. Some programs may also handle more then one File to open.
If you press the **Alt**-key while dropping the files, the menu will start for each file the application. This is useful, because most of the applications, which only can handle one document (f.e. Notepad or Paintbrush), also accept only one file as argument.
Otherwise the files are send as one package to the application.

## Drag'n Drop:
## Insert Files as menu items in a pinned (Sub-)menu

A very comfortable way to install new programs or documents in a submenu is using drag and drop with the File-Manager.

- Open the submenu, which should contain the new menu item and stick it on the desktop by using the pin.
- It is recommended to click a second time on the pin, so it is complete pinned. The pinned menu is now every time in the front of the desktop.
- Open the File-Manager.
- Choose in the File-Manager the file(s) to install and
- drag the file(s) with the left Mousebutton over the menuitem, on which position the program or document should insert.
- Press the **Ctrl**-key to add the new menu item after the position.

*or*

- Press the **Alt**-key to insert the new menu item ahead of the position.

*and*

- drop the file(s).

## The Program-Manager as menu

Comment

There are two modes to include the Program-Manager and its child windows in to the menu structure.
It is possible to bind in the Program-Managers groups **dynamically** in the menu. To do this you have to activate the option **Dynamic PMGs** in the dialog General Properties in the section **Miscellaneous**.

If you edit menuitems in the **dynamically** bound program groups, the changes are also reflected in the Program-Manager itself. If you change an item in the Program-Manager the changes are also showed in the corresponding menuitems.

This dynamically method, to include the Program-Managers Groups in to the menu has the disadvantage, that the icon information had to extract out of the *.GRP-files every time the corresponding submenus are showed.

Because this dynamically method is relative slow, by default all Program-Managers Groups are **converted** in the internal format and saved in the menu file PROGMAN.MNU.

- To include the Program-Manager in a submenu as a menu item, the same steps are necessary, described in Insert a menu item respectively in Add a menu item.
- Select in the Edit dialog the option **Progr.-Manager**. The suggested declarations for **Filename** must, and for **Label** can be accepted.

The connection between the **converted** Program-Managers Groups and the corresponding submenus are not automatically refreshed.
If a installation program had add a group or add an item to an existent group, the corresponding submenus ("Progman" or a submenu) have to be explicit actualized.
To refresh (reread the PROGRAM.INI or a GRP-file) such a submenu you have to hold down the **Shift**-key while opening the submenu at first time. If you changed an item of this submenu, this changes are discarded.

## A single Program-Group as menu

Comment

It is also possible to include a single Program-Group (a sub-window of the Program-Manager) as a submenu in the menu.

- To include a Program-Group in a submenu as a menu item, the same steps are necessary, described in Insert a menu item respectively in Add a menu item.
- Select in the Edit dialog the option **Program-Group**.
- Use **Browse** to pick an existing Group file (*.grp) in your Windows-directory.
  (If you pick Group file not installed in the Program-Manager the menu may be crashed!)

If you haven't activated the option **Dynamic PMGs** in the dialog General Properties in the section **Miscellaneous**, you have to hold down the **Shift**-key while opening the corresponding submenu at the first time. After that the information about the installed programs is written in the PROGMAN.MNU.

(The name of a group (the description) must not begin with a space or be empty.)

**Comment**

I hope the Program-Manager and the Program-Groups are quite transparent implemented, but there are still some problems.
However, you sure recognized, the icons are considerably build slower than in normal submenus.

## Edit menu items of a Program-Group

Comment

!This help page is only relevant, if you activated the option **Dynamic PMGs** in the dialog General Properties in the section **Miscellaneous**.
Otherwise the converted Progman Manager groups have the same behavior as external submenus.!

You can use the normal methods to edit the menu items in a **Program-Group** (also **Program-Manager**) build as submenus.
You have to notice, that all changes (insert, add, edit, delete) will have also an effect to the Windows Program-Manager!! For example: If you delete a menu item in a Program-Group, the installation of the corresponding program in the Windows Program-Manager will also be deleted.

But there are following differences:

- In contrast to normal menu items it doesn't make a different using **Add** or **Insert**. At all: the positions of the menuitems are taken over in an accidental order.
- Of course you can only insert normal menu items (no submenus or else) in a Program-Group. In the Edit dialog this limitation is presented with the option **PM-Item**.

**Comments**

- The icons are supported, but the short-cuts and the option starting programs icon sized will have no effects.

## Menus with a 3-D-Effect

Comment

Instead of the normal appearance of Windows menus you can supply the menus with a 3-D-Effect (At first it was only a fashion, it becomes more and more a standard. The programmers do anything for the users! ;).
Than you can also customize the Font, using by the menu.

- Select in the dialog General Properties the option **Graphic**.

If you activated this option 4 different Styles are available.

## Choosing the menu-Font

Comment

To change the menu font, it is necessary to check the Option Graphic.

- Select in the dialog General Properties the button **Fonts** and choose in the Windows standard dialog your preferenced font.

## Use icons in the menus

Comment
Comment 2

One of the highlights of the menu is the feature, showing programs and Documents automatically by their icons. For programs, which haven't an icon, like DOS-applications, default icons are used.
The first call needs a little bit more time to extract the icon and write it in the database (RRKDAT.xxx and RRKIDX.xxx in the directory of RRKMENU.EXE). But normally the appearance is quite fast.
If you have a very slow computer you may deactivate this option.

- Select in the dialog General Properties the option **icons**.
- In the field **icon Size** you can choose the preferenced icon size. 32 pixels is the normal size, also showed by the Program-Manager. 24 and 16 pixels are equivalent smaller.

**Comment**

If you change the Font- or the icon size, while pinned menus exist on the desktop, you can refresh this pinned menus with the **F12**-key. (This may be fixed in a later version.)

## Starting Options

**As Shell:**
Normally Windows starts at first the Program-Manager for Windows Shell.
If you like the menu, and find the Program-Manager unnecessary for you, you can use the menu as Windows Shell:
(Comment: At my experience the menu is more economically with the System resources as the Program-Manager.)

- Select in the dialog General Properties in section **Miscellaneous** the option **As Shell** and restart Windows.

You can enjoy the purity starting Windows with a complete empty desktop (All wishes to order and purity tend to negate all things to nothing :-), and like a deus ex machina the menu waits for you in the background.
But the menu also supports a submenu, called **Autostart** or **Startup**. It functions in the same way as in the Program-Manager of Windows. Create this submenu, if you want start applications at the beginning of Windows. The menu also supports the **run** and **load** directives in the WIN.INI.

**Save the Desktop by ending the menu**
If you have collected our preferences Applications and our Project-Files (may realized with **Searching Directories**.) in pinned submenus on the desktop, you can induce the menu to restore these pinned submenus by restarting the menu.
submenus, build by **Drive Directories,** will not be restored.

- Select in the dialog General Properties in section **Miscellaneous** the option **Save DT.**

## Using the middle Mousebutton

Although the Microsoft Mouse has only 2 mousebuttons, 3 mousebuttons are supported by Windows. (It depends also by the used mouse driver)
If you have a mouse with 3 mousebuttons, it is recommended using the 3rd, middle mousebutton to popup the menu.

- Unfortunately you can't test the 3rd mousebutton in the Windows Control. For precaution at first it is recommended to unhide the menu icon (the black arrow) by using **Hide icon** in the submenu **Menu Setup** or stick the submenu **Menu Setup** on the desktop.
- Select in the dialog General Properties in the section **Mouse** the option **middle Mousebutton**.
- Test the middle mousebutton.

- If your Mouse doesn't support the 3rd mousebutton, you can use the emulation of the middle mousebutton.

Look also: Avoid Conflicts with other programs

## Improve the Speed of the menu

- You have to install a disk cache (smartdrv or similar), to use the menu with an acceptable speed.
- Don't use the Program-Manager submenus. They are much slower, as the other menu items.

If you haven't serious problems with our RAM (286 / 1 MB) you can improve the speed of the menu by

- selecting in the dialog General Properties in section **Miscellaneous** the option **Index in RAM**.

Without this option the menu needs permanently only ~25 KB non-discardable RAM, with this option ~ 100 KB more (It depends by the number of Document-Associations and registered icons in the menu-Database).

Commend

## Avoid Conflicts with other programs

The NMfW is a kind of TSR, hooking the mouse actions.
Unimportant which application is actual active, pressing the right (middle) Mousebutton the menu comes up.
This may be causes problems.
To minimize these problems, the menu is very jealous of other programs using the right (middle) mousebuttons. More and more programs supports the right mousebutton for local edit options. In this case, you normally can't access this features of the programs, because every time the menu is popped-up.

- If you have a mouse with 3 mousebuttons, it is simple to solve this problem: Use the middle Mousebutton for the menu!

- Otherwise can use the Option **Only Desktop** in the submenu **Menu Setup**. Then the menu only will popped-up, when the mousecursor is over the background of the desktop. In the applications the right mousebutton will function in the normal way.
- An other way to disable the right mousebutton for the menu, is to hold down the **Shift**-key while pushing the right mousebutton.

If you select the option **Hotkeys** in section **Menu - Hooks** in the dialog General Properties, you can still popup the menu independent of the position of the mousecursor by pressing the Alt- and Pause- Key.

**Comment**
This option is activated by default.

## Hide the program icon

One of the aim of the menu is, to clean up the overcrowded Desktop. By consequence you can also hide the program icon of the menu (a black arrow).
*Either:*

- Select in the System menu of the program icon the menuitem **hide icon**.

*or:*

- Select in the submenu **Menu Setup** the menu item **hide icon**.

To unhide the icon:

- Of course you can unhide the icon only with the menu item in the submenu **Menu Setup**.

 visible.
 hidden

## only Desktop

If you want, that the menu will only popped-up, when the mousecursor is over the background of the Desktop, you can select **only Desktop** in the submenu **Menu Setup**.


1. The menu will popped-up on every place of the desktop.


2. The menu will only popped-up, if the mousecursor is over the background of the desktop.


3. The menu will only popped-up, if the mousecursor is over the background of the desktop **or** if the mousecursor if over a title bar of a window

You can toggle between 2 and 3 with pressing **Ctrl** while selecting **only Desktop**.

Look also: Avoid Conflicts with other programs

## Using Hotkeys

To launch programs, open a submenu or execute Internal Commands with the keyboard, you can use Hotkeys. In contrast to ShortCuts for Hotkey it is not necessary, that a menu is popped-up.

If you want use Hotkeys, the option **Hotkeys** in the section **Menu - Hooks** in the dialog General Properties must be activated. It is also recommended that the option **Index in RAM** in the section **Miscellaneous** is activated, otherwise the reaction of the keyboard may be slower.

You can define Hotkeys in two ways:

**1. In the Edit-Dialog:**
If you define a new menu item or edit an existent menu item, you can also define a Hotkey in the editbox **Hotkey** to execute the command line with a key combination.
If there is a checkbox **Global** (otherwise you declare a ShortCut) you have to check it.
If the editbox Hotkey has the focus, push the key combination.
If for command line (also the parameter, working directory, window size and the icon is relevant) a Hotkey is defined, the key combination is showed in the editbox Hotkey. If you declare a new Hotkey, and the key combination is already defined for another command line, the old association will be overwrite.

**2. In the Dialog Hotkeys:**
If you want to have an overview of the defined Hotkeys, you can use the Dialog Hotkeys to define and undefine Hotkeys.

## Using Context-Sensitive menus

Context-Sensitive menus extent the functionality or the comfort of access of a defined Window-Class like EditBoxes, ListBoxes or the DOS-Box. You can get the associated context sensitive menu with the right mousebutton, if the mouse cursor is over a defined Window-Class.

If you want you the context sensitive menus, the option **Context-sensitive Menus** in the dialog General Properties must be activated.

For a description of the context senstive menus see at:
Context-sensitive Add-Ons for new Menus for Windows

## Popup the local Editmenu in pinned Menus

It is possible to edit the contents of a Menu which is stuck to the desktop using mouse functions. After sticking the menu to the desktop, you can choose an item on it by using the right (middle) mouse button instead of the left button and an edit menu will pop up. This menu will allow you to edit the highlighted item, insert or add new items, or delete the item.

See also: Editmenu of pinned Menus

## Using the Menu without a Mouse.

It is possible to use the Menu without a mouse. And even on a system with a mouse, some of the keyboard shortcuts below may make it easier for you to move quickly through the menus.

The following hot-keys are active for New Menus for Windows:

- **Alt**-**Pause** will popup the Menu just like the right (middle) mouse button.
- **Esc** will close the current Menu or SubMenu. It may be necessary to hit **Esc** several times if you wish to back out of the Menu system completely. Menus which have been stuck to the desktop cannot be closed with **Esc**.
- Use the **Up** and **Down Arrows** on the Keyboard to highlight a menu item.
- Use the **Right Arrow** or the **Return** key on the keyboard to open the highlighted Submenu.
- Use the **Left Arrow** or the **Esc** key to close the current Submenu.
- To launch a program use the **Return** key.
- To stick a Menu, highlight the pin head with the cursor keys, and hit the **Return**-key. When you first open a menu or sub-menu the pin head is highlighted by default. The pin head is at the top of the just above the first menu item, you can get there with the cursor keys.

The following functions can only be used when a pinned menu is the focus.

- **Shift-PgUp** will roll up the pinned Menu leaving only the Menu heading visible.
- **Shift-PgDown** will roll down the pinned Menu, revealing all the items on the Menu.
- **Shift-Esc** will hold a pinned menu on top of all other programs. This has the same effect as clicking a second time on the pin head with the mouse.
- Pressing the **Shift-Esc** keys again when the menu is being held on top will close the menu. (In order to close a menu that is not held on top you need to hit **Shift-Esc** twice).
- **F9** will arrange all the pinned Menus along the left side of the desktop.
- **F11** will arrange all the pinned Menus along the top of the desktop.
- Using the **Arrow** keys while holding down **Shift** will move a pinned Menu.

## External Menus

In addition to defining your own sub-menus in the RRKMENU.INI file you can define menus in external .MNU files. These files are defined in the same format as the sub-menus in RRKMENU.INI. To add one of these menu files to your Menu, simply add an item, set the Label that you want displayed for the menu entry, and set the filename to the filename of your .MNU file, and declare it as a submenu item.

You can store .MNU files anywhere on your system. If you do not specify a path for the .MNU filename, New Menus for Windows will first look for them in the same directory as RRKMENU.INI, then search the Windows directory, and finally search the directories in the PATH environment variable.

**Advantages to using external menus:**

Using external menu files can be very helpful in multi-user network environments. Each user could have their individual RRKMENU.INI file with a sub-menu linking to a menu located on a server that would be common to all users. This allows the network administrator to easily manage a menu system for a large group of people while each individual user can still modify his/her own local menu to some extent.

One additional feature for multi-user environments is the ability to lock Submenu's. If the submenu is defined in the .INI file with the type 0=8, the user cannot change or delete the submenu. The user can only remove the link to that sub-menu. If a user wanted his/her own modified menu, they could copy the appropriate .MNU file to their local directory, delete the link to the network sub-menu, and add a link to their own modified version.

See also: Install the Menu in a Net environment

**Note:** In the standard configuration of the Menu some Submenus (i.e. "Windows-Manager") are actually external menus (in this case ADDONS.MNU).

## Declare Icons for Non-Windows Programs

DOS-programs, PIF, COM and BAT-Files don't have individual Icons like their Windows counterparts. These programs will be displayed with generic standard icons.

It is possible to define your own icons for DOS applications to display on the Menu. There are two methods that can be used:

The first way is to declare an Icon in the edit dialog. To do this, edit the menu item for which you wish to change the icon. This is accomplished by highlighting the menu item and pressing F6, or if the menu has been stuck to the desktop, clicking on the item with the right (middle) mouse button and choosing the **edit** option from the edit-menu. Next, select Icons, and specify an icon-file (.ICO) to use.

This method has the disadvantage that if you change the icon size, color resolution, or menu style in the general Properties dialog, any custom defined icons will be lost, and you will have to define them again.

The other method is to take the icon (.ICO) file you were intending to use and make a copy of it with the same name as the application filename in the same directory as the application. For example, if you wanted to create an icon for the program `dosapplic.exe`, you would create the icon file `dosapplic.ico`, and place it in the same directory as `dosapplic.exe`.

After doing this, the icon should show up alongside the menu item pointing to that file. If you don't see the results on the menu after putting an icon file into the appropriate directory, then delete the files `RRKDAT.*` and `RRKIDX.*` in the New Menus for Windows directory. These files are used for caching icon graphics.

## Using ShortCuts and Hotkeys

It is possible to define ShortCuts, and Hotkeys for your submenus and applications. ShortCuts are Ctrl key combinations that only operate when the menu is open on the desktop. Hotkeys on the other hand can be activated at any time whether the menu is activated or not and can be assigned to any submenu or application.

There are two ways to define a ShortCut for a Submenu: You can edit a Submenu entry in the Edit-dialog or you can use the dialog ShortCuts to define new ShortCuts or delete existing ShortCuts.

You can also define the Hotkey for a particular item in the Edit-dialog by setting the Hotkey and checking the **Global** box, or by using the Hotkeys dialog to assign Hotkeys throughout the system.

## Use the right Mousebutton with a Second (Sub-)Menu

If you normally use the **middle** mousebutton to popup the Menu, you can declare a second (sub-)Menu for your **right** mousebutton. You might use this to normally activate your menu with the middle button, but use the right button to activate a special tools menu, or configuration options.

- Open the dialog **General Properties** and change to the section **Mouse**.
- Use **Browse** to pick a Submenu from the list of available menus.

You can also start an application or an internal command with a single right mousebutton click, but you have to fill in the field **Submenu for the right mousebutton** with a valid INI-line.

## Change the style of the Menu

In the General Properties dialog you can choose one of four graphical styles for the Menu.

**Standard**
: This is a simple style, the menuitems are small and the menus are updated on the screen quickly.

**OpenLook**
: An imitation of the OpenLook environment available for UNIX machines using the X-Windows GUI of the same name.

**Beta**
: Just a one-evening-production.

**Motif**
A style drawing from the X/Motif window manager available for UNIX X-Windows based machines.

If you change the style to or from **Standard** all icon database files RRKIDX.* and RRKDAT.* are deleted. These files store the information for custom defined icons for DOS applications.

## Using a Submenu as a Desktop-Panel

It is possible to install the your favorite and important programs into one or more submenus and stick these Submenus on your desktop. You can also toggle the menus to display the icons only, text only, or both (the default).

**Note**: See Drag'n Drop: Drag'n Drop: ... in Pinned Menus for information on working with pinned Menus.

You can set how the menu is displayed in the edit dialog. The options are Normal, Icon, and Text. Normal will display icons (if they are enabled in the General Properties dialog), and the text description for each menu entry. The Icon option displays only icons, and the Text option displays only the menu text.

**Using Drag'n Drop-Panels:**

Because the title bars of menus which only display icons are to small to show the pin and the Roll-Up-Button, there have been some special functions added to make them accessible. When first activated the panel looks like this:

In this state you can roll up and down and move the panel with the mouse. After pressing the **Tab** key, the panel will look like:

Now you can only move the panel with the mouse. Pressing **Tab** again will make the panel look like:

In this state you can click once on the pin to hold the pinned Menu on the top, or twice to close the menu. Pressing **Tab** at this point brings you back to the first state of the menu.

Of course the keyboard ShortCuts defined in Using the Menu without a Mouse always work in the normal way.

## Running a Windows incompatible DOS Program

Some DOS programs just won't run correctly under Windows. Often this is because they use their own DOS-Extender or the operate in protected mode. Autocad and 3D Studio (the non-windows versions), and some games are good examples of these types of programs. Some programs like disk utilities need direct access to the hardware that Windows doesn't allow.

Unfortunately, there is a way through New Menus for Windows that you can run programs that act in such a manner, they just cannot be run through windows. However, if you have programs which need direct video access, or need to run by themselves to be fast enough (i.e. many games) you can quit Windows, execute one DOS-program and restart Windows with a menu entry.
To do this, create a menu item in the submenu where you want the program to reside. In the **command line** of the Edit dialog, enter the command:

```
MNU_WIN_REBOOT c:\command.com /c c:\my_dir\my_progr my_parameters
```

This might look a little much at first, but its really pretty simple:

> The first parameter on the line (`MNU_WIN_REBOOT`) is the Menu command to reboot windows.
> The second parameter (`c:\command.com`) is the name of the DOS command interpreter. If you are using a different command interpreter (i.e. NDOS, or 4DOS, you should change this accordingly).
> The `/c` option is to tell the command interpreter to run the following program, and then return to New Menus for Windows control.
> The parameter after that (`c:\my_dir\my_progr`) is the filename of the program you want to run.
> And the `my_parameters` option is any parameters you wish to pass to that program.

Set the **Label** for the program to the text you want displayed on the menu enter, and you can set a custom icon if you desire. Now when you select this menu-item, Windows will shut-down, run the specified program, and reload once the program has finished executing.

If you only use the `MNU_WIN_REBOOT` command without additional parameters, Windows will simply quit and restart when you select this menu entry. If you hold down the **Shift**-key when selecting this menu item the system (computer) will be rebooted.

# The Dialog - Select Submenu

## How to get the dialog

In the Edit-Dialog or in the dialog General Properties, you have the **browse**-button to get this subdialog.

## Explanations

In this dialog you can select a submenu.
Submenus are defined in the menu files. By default the file extensions for menu files are *.MNU. Only the special case of the RRKMENU.INI you should this this file extension.

In the edit field at the top of the dialog you the current menu file is showed. If you want to pick another one, you can click onto the little yellow folder.
Below you can see a list of all submenus, which are defined in the current submenu.
To select a submenu, click onto the item in the list and select the button OK.

# The Dialog - Select Icon

### How to get the dialog
Select the **Icon...** button in the Menu - Edit dialog**.**

### Explanations
After choosing an Iconfile in the file-dialog, all Icons of the file are listed. After selecting one of these Icons the program, declared in the Editdialog will be shown with this Icon.
If you have declared a document in the Editdialog, the Icon change will not have any effect in the Menu (its icon can only be the icon of its associated executable).

### Comments
If you change the size of the icons shown in the menu, or if you change the number of colors under Windows, the Icons assigned to programs shown in the Menu are discarded, and recreated when accessed by the Menu again.
Another way to assign non-Windows (DOS) programs with an Icon (independent of the icon size and the color-resoulution) is described in Declare Icons for Non-Windows Programs.

### Icons
Here, all Icons of the selected Iconfile are listed. The black triangle shows the selected Icon.

### Browse
Opens a Standard-Dialog to choose an Iconfile (*.ICO;*.EXE;*.DLL).

### OK
The new Icon for the program is only re-assigned, when you confirm it in the Editdialog

### Cancel
Closes the dialog without defining a new Icon

## The Dialog - ShortCuts for Submenus

### How to get the Dialog
Select the **ShortCuts** button in the Configuration dialog.

### Explanations
If you popped-up the Menu you can, then jump to a Submenu with a Hotkey.
These Hotkeys can be declared in this dialog.
See also: Using ShortCuts.

### Comments
It is also possible to declare the right Mousebutton with a second Submenu.

### Ctrl - Hotkeys
Here, all Hotkeys assignments in the menu are listed. Click on an item to look at the associated Submenu.

### Submenus
Here, all Submenus, defined in RRKMENU.INI, are listed.

### Associate
If you type the **Ctrl-** key with a char as the Hotkey and you have selected a Submenu you can associate the Hotkey with the Submenu.

### Detach
Select a Hotkey-char (the associated Submenu will automatically be selected) and push **Detach**, to detach the Association.

## The Dialog Hotkeys

### How to get the Dialog
Select the **Hotkeys** button in the dialog Configuration.

### Explanations
In this dialog all key combinations for Hotkeys, which are associated with a command line, are listed.
It is also possible to assign Hotkeys for a Menuitem in the Edit - dialog.

In the field **Commandline** place the command line -including the parameters, the working directory and a valid window-size. You can also declare an Internal Command as a command line.

You can define a new key combination for a Hotkey by placing the cursor in the **KeyComb.:** field and then typing the desired keys.

### Comments
It is also possible to assign Hotkeys to a Menuitem in the Edit - dialog. But it might be better to first review all of the already-assigned Hotkeys, because the command line in the - Hotkey - association for a menuitem may be ambiguous.

### KeyCombination
Here, all defined Hotkeys are listed.

### Command Line
Here, all command lines, which are associated with a Hotkey, are listed.

### Associate
Once you have declared a **key combination** for a new Hotkey and a command line (maybe via **Browse**) you should then use **Associate** to assign the new Hotkey.

### Detach
If you want to unassigne a Hotkey, then select the **Key Combination** and click on **Detach**.

### Hotkey

With a Hotkey (a key combination with the Alt and/or Ctrl- and the Shift-key with a char) you can launch Programs or execute Internal Command with the keyboard at any time.
For a valid Hotkey the Alt- or the Ctrl-key and a char must be used (the Shift-key can be used in addition, but not alone).
Examples of valid Hotkeys key combinations:
`Ctrl+D; Alt+1; Alt+Shift+T; Ctrl+Shift+Z; Alt+Ctrl+Shift+O`
**Not valid** are following combinations:
`Shift+A; Ctrl+#`
The Hotkeys only function, if in the dialog general properties the option **Alt+Pause-Popup (Hotkeys)** is activated.

## The API

## Internal Commands

Internal commands begins allways the the string MNU_
Variables beginn with the char @.

**@FM_ARGS**
Variable, which is only valid in a context senstive menu for the File-Manager.
If you want to add in the context sensitive menu for **File-Manager**, you can use the @FM_ARGS as a placeholder for the hilighted files in the File-Manager.
4=0|WinWord|C:\WW\WINWORD.EXE @FM_ARG
starts WinWord and load the files highlighted in the File-Manager.

**@HWND**
Variable, which is only valid in a context sensitive menu.
Serves the handle of the window, which is hooked via a context sensitive menu. This variable can be used for API-functions.
See also: Call a DLL-Function

**@RRKMENU_INI**
This variable serves a string, with the file name (including path) to the RRKMENU.INI.

**MNU_ADD**
It doesn't make sense to use this internal command in another place, as in the Edit-Menu of pinned Menus.

**MNU_ARRANGE_LEFT**
Minimizes all sticked Menus - except the Menus only with icons and the pinned Drop-Down Menus - and arranges them at the left side of the desktop.
See also: Arrange Icons, Tile Windows vert., Tile Windows hor. and Cascade Windows
and: Arrange Menus at Top and Arrange Menus at Left

**MNU_ARRANGE_TOP**
Minimizes all sticked Menus - except the Menus only with icons and the pinned Drop-Down Menus - and arranges them at the top of the desktop.
See also: Arrange Icons, Tile Windows vert., Tile Windows hor. and Cascade Windows
and: Arrange Menus at Top and Arrange Menus at Left

**MNU_BACKGROUND**
See Only Background

**MNU_CALLDLL**
Calls an external DLL-function.
See also: Call a DLL-Function

**MNU_RUN**
This Menuitem prompts you for a commandline.
In the dialog Menu - Execute you can declare a command line.

**MNU_CONFIG**
Calls the dialog Configuration.

**MNU_DEL**
It doesn't make a sense, to use this internal command at an other place, as in the Edit-Menu of pinned Menus.

**MNU_DEL**
Declares a Printerdevice.

**MNU_DOWITHALLWIN**
Only for internal purposes
Executes a function, which is declared with a number:
MNU_DOWITHALLWIN 1: Minimizes all visible windows.
MNU_DOWITHALLWIN 2: Closes all visible windows.
See also: Save All, Minimize All and Close All

**MNU_EDIT**
It doesn't make a sense, to use this internal command in another place, as in the Edit-Menu of pinned Menus.

**MNU_END**
Ends the Menu or Windows; with or without confirmation.

MNU_END 1
Allways shoot down Windows.
See also: End

**MNU_HIDEICON**
Hide Icon

**MNU_INFO**
Calls the info. dialog.
See also: Info

**MNU_INS**
It doesn't make a sense, to use this internal command in another place, as in the Edit-Menu of pinned Menus.

**MNU_KB**
Send keyboard actions to the active window.
See also: Generate keyboard actions with a Menuitem

**MNU_MBATCH**
Executes all Menuitem in a given Submenu as batch.
See also: Execute a Submenu as Batch

**MNU_MOUSE_CAP**
Only for internal purposes
Capture the mouse, serves the window, the user clicked with the left mouse button and execute the function, which is declared as number:
MNU_MOUSE_CAP 1: To kill the Window
See also: Kill

MNU_MOUSE_CAP 10: Add a window to saved desktop (See also MNU_SDT)
MNU_MOUSE_CAP 11: Delete a window from the saved desktop (See also MNU_SDT).
See also: Save Wins DT, AddDT and DelDT

**MNU_MTW**
Sends a message to a window.
See also: Send Messages to a Window

**MNU_PRINTERS**
Generate a submenu with all printers installed in Windows.
See also: Printer

**MNU_PROGMAN**
Generate a submenu with all Program-Groups of the Program-Manager.
See also. The Program Manager as Menu

**MNU_SDT**
With this command you can save all positions of the applications (windows) on the desktop.
See also: Save Wins DT, AddDT and DelDT

**MNU_SETUP**
It is only used to declare a special icon for the submenu Menu Setup.
See also: Setup

**MNU_SWITCHTOSCREEN C R**
To use this internal command the Virtual Desktop (see also MNU_VSCREEN below) must be activated.
C = column; R = row.
Switches to the virtual desktop, which is declared with C and R. The first upper left virtual desktop as the number C=0 and R=0.

MNU_SWITCHTOSCREEN 1 2
This command line switches to the virtual desktop in the second column and the third row.
If the number of column or row is higher as 8, the virtual desktop will be restored. This means, that all windows will be moved into the first upper left virtual desktop.

**MNU_TASKS**
Generate a Submenu with all Windows (programs), the free memory and System-Resources are also shown.

Normally the syntax for MNU_TASKS in a Menu item is:
4=1|Tasks|MNU_TASKS

But it is also possible to declare a Submenu in this way:
[My Tasks-Submenu]
1=MNU_TASKS

See also: Tasks

**MNU_WIN_REBOOT**
Shoot down Windows without confirmation and restart windows.

**MNU_WIN_REBOOT 1**
Your computer will reboot (similar to Ctrl-Alt-Del).
Alternatively you can press the **Shift**-key, while selecting a **MNU_WIN_REBOOT**-Menuitem,.

You can also execute a DOS-Program or DOS-command.
**If you want to execute a DOS-Program it is important to declare a command-interpreter (normally command.com). Otherwise the system halts!**
A command-line can look like:
`MNU_WIN_REBOOT c:\command.com /c c:\my_dir\myprogram`
`c:\command.com` is the command-interpreter of DOS. `/c` tells the command.com to execute one program `c:\mein_dir\meinprog` and after that, to terminate itself.

This option is useful to execute a DOS-Program, which can not run in a DOS-Box of Windows.
If you don't declare parameters after MNU_WIN_REBOOT, Windows will restart with no action on the DOS-level.

With the command
**MNU_WIN_REBOOT 2**
you can define the DOS command line in a dialog.

See also: Reboot

**MNU_VSCREEN**
Switches the Virtual Desktop on and off.

**NIX**
is a short form of the German word 'Nothing' and is used for submenus, which don't have internal commands or which aren't stored in external menu (*.MNU) files. In the version 1.3 this keyword is obsolote. Instead the default file name for submenus stored in the RRKMENU.INI is RRKMENU.INI

**Unsinn**
is a German word for nonsense. In older versions of New Menus f. W. I used this word as a debug-default.

## Commandline options

Following command line options for the RRKMENU.EXE are available:

**/menupath c:\my_path\to\NMfW_INI**
or
**-menupath c:\my_path\to\NMfW_INI**
Sets the default path, where your RRKMENU.INI can be found.
See also: Install the Menu in a Net environment

**/i 0|Mylabel|mycommand**
or
**-i 0|Mylabel|mycommand**

Executed a iniline, but only if NMfW is still running, otherwise NMfW will be started and the command line options will be ignored.
The command line:

```
X:\MYPATH\RRKMENU.EXE /i 1|Tasks|MNU_TASKS<2
```

pops up the Submenu **Tasks** with no title bar.

default
The parameter will be interpreted as a program. NMfW try to start the program.

By the way:
You can also start Windows and NMfW from DOS just starting RRKMENU.EXE from the DOS-Prompt.

## Calling a Menu from another application or via the commandline

You can call a menu from another program.
To do this, you have to call the RRKMENU.EXE pass after the switch /i a complete INI line as parameter:
c:\mydir\RRKMENU.EXE /i 1|MySubMenu|NIX
A condition is, that the RRKMENU.EXE is still running.
If you want to replace the Windows **Tasks Manager** with the tasks menu, you can insert in the section [boot] of the SYSTEM.INI following line:
[boot]
...
taskman.exe=C:\MYPATH\RRKMENU.EXE /i 1|Tasks|MNU_TASKS;
Edit the SYSTEM.INI in your Windows directory.
[boot]
taskman.exe=c:\mypath\RRKMENU.EXE /i 1|Tasks|MNU_TASKS;
(This should be one line. The semicolon ";" at the end of the line is only important, if you use this line in the SYSTEM.INI.)
Now you get the Submenu "Tasks" if you doubleclick onto the desktop.

## Simulate keyboard actions with a Menu item

If you want generate system-wide keyboard actions with a menu item, you can use the internal command **MNU_KB**.

MNU_KB Char,Schiftstatus secondChar,secondSchiftstatus ...

- Char:
  The key (a char or a VK_scancode declared in windows.h).
- Schiftstatus:
  Shift-, Alt- and/or Ctrl-key as bit-combination.
  - 0 = no Shift
  - 1 = Shift-key
  - 2 = Ctrl-key
  - 4 = Alt -key

  The Shift-Ctrl-Alt-combinations results out of the sum of the bits. E.g.: to produce the key-combination d-Alt-Ctrl, you have to declare `D,6` .
- The single key-statements are separated with a whitespace.

If you want pass long keyboard sequences to an application, you may use a Submenu as batch

## Execute a Submenu as batch

You can execute a whole Submenu as a batch. To do this you have to define a normal menuitem and as command line:
MNU_MBATCH Submenu[,menu file]
The string for the menu file is only necessary, if the submenu is not declared in the RRKMENU.INI.
All normal menuitems are executes. Submenus, Searching Directories and Drive Directories are ignored.
In a Submenu, which is executed as a batch, also subcalls to other submenus via MNU_MBATCH are allowed, but be care of recursive calls.
E.g. you can start Write with following batch:
x=0|Write|MNU_MBATCH Start Write,MACROS.MNU;WRITE.EXE
(Label = Write; command line = MNU_MBATCH Start Write,MACROS.MNU; Icon = Write.exe)
and in the file MACROS.MNU following submenu is defined:
[Start Write]
1=0|Write|WRITE.EXE
2=0|keyboard action|MNU_KB C,4 R,0
In the first line Write will be launched.
In the second line, a keyboard macro Alt+C R will be executed (I'm not sure, if this are the right keys. Look at the menu of Write for the correct accelerators!), what should be have the effect, that Write will start with the visible ruler.

## Call a DLL-Function

It is also possible to call a function you have wrote, in a DLL or call a system function of the Windows API with a menu item.
The syntax of a command line of a menu item:

MNU_CALLDLL Modulname Functionname Parameter

- **MNU_CALLDLL** is the internal command to point to function in Pascal-convention.

- **Modulname** is the name of the DLL, e.g. `my_own.dll` (the path to the DLL may be required.)
  The most used modules of the Windows-API are USER, GDI and KERNEL.
- **Functionname** is either the name of the function or its exported ordinal-number.
- **Parameter** is a list of the parameters for the function. (The parameters are passed in the Pascal-convention.)
  For the type of the Parameters you have to consider following conventions:
  - Integer number are written as normal numbers. Negative numbers get an minus sign (e.g. `123` or `-321`)
    If you want write an number in hex you have to begin the number with 0X (null-x): e.g. `0X1A4C`)
  - Long Integer (long) ends with a L (`1234L` or `65L`)
  - Floating point numbers are written with a point (`2.14` or `12.0`).
  - Bytes (a char) are started with a inverted comma ' (`'c` or `'x`).
  - char strings are enclosed with quotation marks (`"hello"`).

**Warning!!**
**Be careful with the parameters. If you declare the wrong type or numbers of parameters you get an GP-fault!**

*Examples*:

A beeping Menu item:
```
MNU_CALLDLL USER MESSAGEBEEP 0
```

A Menu item calls a MessageBox:
```
MNU_CALLDLL USER MESSAGEBOX 0 "And good Bye" "Hello" 4
```

Even undocumented functions of Windows can be called:

A Menu item to arrange all minimized Windows at the bottom of the desktop:
```
MNU_CALDLL USER ARRANGEICONICWINDOWS 0XE8C
```
(`0XE8C` is the HWND of the desktop)

Arrange all not icon sized Windows next to each other:
```
MNU_CALLDLL USER TILECHILDWINDOWS 0XE8C
```

Arrange all not icon sized Windows overlapping:
```
MNU_CALLDLL USER CASCADECHILDWINDOWS 0XE8C 0
```

The command:
```
MNU_CALLDLL USER SENDMESSAGE 0XFFFF 0X0011 0 0L
```
with `0XFFFF`=HWND_BROADCAST and `0X0011`=WM_QUERYENDSESSION,
has the function to ask all win-programs, if they have any unsaved changes.
(You can add this command line to the Submenu `Menu Setup` if you find it is useful.)

The Label of a Menu item, with MNU_CALLDLL in the command line, is ad libitum, you can also declare another icon.

## Associate a Window-Class with a Menu

New Menus for Windows is not only a Windows-Shell. It also possesses some features to enhance Windows. If you can write a DLL and/or you know some details of Windows-Messages, you can customize and enhance the Windows-functionality of Windows standard elements (like EditBox, ListBox or Window-Classes of your application) with a pop up menu. If you want participate to New Menus for Windows with your application, you haven't change any line of your code.
How this works, was demonstrated with the Window Classes EditBox (Notepad), ListBox, ScrollBar an tty (the DOS-Box in the 386-modus).
If the option **Context-Sensitive Menus** in the dialog General Properties is activated, the Window Classes, listed in the section `[WinClasses]`, are customized with a local Menu. This Menu can be popped up with the right mouse button if the mouse is over a registered class-window.
In the pa
In the pages Context-sensitive Add-Ons for New Menus for Windows the defined context sensitve menus are documented

In the command line of a menu item you can call via MNU_CALLDLL any function of DLL. You can use the expression @HWND for the Handle of class window as a parameter to a function.
If you only want to send a message (via `PostMessage(...)`) to this window you can use the internal command MNU_MTW.
It is also possible to send keyboard events to a window with MNU_KB.

If you have some ideas to enhance Windows functionality or if you want to use Menus for your applications, please contact me.

## Send Messages to a Window

With the internal command **MNU_MTW** (Message to window) in a command line of a Windows Class Menu you can send Messages to the Class-Window. (Of course it would also function with: `MNU_CALLDLL USER POSTMESSAGE @HWND` (and so on.)).

MNU_MTW Parent message wParam lParam

- MNU_MTW sends the message with `PostMessage` to the **Parent** of the Window. This means if Parent=0, the message will be sent to @HWND (the Class Window). If Parent=1 the message will be sent to `GetParent(@HWND)`, if Parent=2 to `GetParent(GetParent(@HWND))` and so on.
- **message** is the message to the window (almost WM_- constants) as UINT.
- **wParam** is the first parameter of the message as WORD.
- **lParam** is the second parameter of the message as LONG. (Exceptionally you don't have to sign this LONG with a 'L' after the number).

You can also declare **message, wParam** and **lParam** as hexadecimal numbers (f.e.: `0X1234`).

## Stick normal Drop-Down-Menus with a Pin

This feature is only available, if in the dialog General Properties the option **HookMenus** is activated.
If you select a Menu from a normal program with a menu bar not with left, but with right mouse button, instead of the normal menu a Menu know by New Menus for Windows appears.
E.g.: If you select the menu 'File' of Write with the right mouse button, you can stick this menu with a needle on the desktop. The pinned Menu stays in the foreground every time. But if you change the focus to an other application the pinned Menus will be hidden.

**Attention**:
The Menus of an application can be changed (e.g. the menu 'Window'); so, if the pinned Menus are not up to date they will send wrong commands to the application. The Menus are refreshed after each command, but nevertheless it is not very safe.
This may cause unexpected reactions and effects. The menu items which are not accessible are showed dimmed, but you can still execute them. Be careful with this.

## Virtual Desktop

Imagine you can have a screen, four or six times bigger than your physical screen, and that you can group your windows freely on this screen.

With the menu item **Window-Manager/Virt. Desktop** you can enable or disable the Virtual Desktop. If you enable it, you can see a small window, which represents the entire virtual screen.

**Dividing the Virtual Desktop**
By default, the virtual desktop is divided into 2 rows by 2 cols. You can see this in the virtual screen window by the two lines, which divide the virtual screen into 4 sections. If you wish to have a bigger virtual screen (with more sections), you can customize it in the **Virtual Desktop** section of the **General Properties** dialog.

**Drag applications windows from one section of the Virtual Desktop to another**
Hold a window with the **left mouse button** in the virtual screen and drag it to another virtual desktop.

**Switch between virtual desktops**
By clicking with the **right mouse button** in a virtual desktop in the virtual screen, you can switch to that virtual desktop.
If you use the right mouse button to pop up the menu, you must define following line in the section [WinClasses] of your RRKMENU.INI.
VScreen=Excluded
Otherwise you can't switch to another virtual screen with the right mouse button.
You can also do this, at any time, with the keyboard. Use **Ctrl+Alt+ F1** ... **Ctrl+Alt+F4** to switch between the first four (2 rows * 2 columns) virtual desktops.
The option **Hotkeys** in the **General Properties** dialog must be activated, so that these keystrokes can be used. Besides, the following lines in the [ProgrammHooks] section of the RRKMENU.INI file must be defined:
393328=0|C0 R0|MNU_SWITCHTOSCREEN 0 0
393329=0|C0 R1|MNU_SWITCHTOSCREEN 1 0
393330=0|C1 R0|MNU_SWITCHTOSCREEN 0 1
393331=0|C1 R1|MNU_SWITCHTOSCREEN 1 1

Additionally you can use the corresponding menu items in the submenu **Window-Manager/Switch to VD**.
With the menu item **restore** in this same submenu, you can switch to the first virtual desktop and move all windows to this desktop.

**Activate an application window**
If you **double-click** with the **left mouse button** in the image of an applications window in the virtual screen, youll activate the application.
In the **Virtual Desktop** section of the **General Properties** dialog you can choose the option **Switch to active window**, to switch automatically to the virtual desktop where the applications window resides.

**Enhancement of the system menus**
If you want one program to appear in all virtual desktops (maybe the clock, because you always want to be able to read the time), click in the system menu of the program with the right mouse button and select the item **Fixed**. Selecting **Moveable** will make the application window stay in the current virtual desktop.

**Close the Virtual Desktop**
If you close the Virtual Desktop with the menu item **Window-Manager/Virt. Desktop**, the first virtual

desktop will be activated and all programs windows will be moved to this virtual desktop.

**Handling of the virtual screen**
You can switch between showing and hiding the **title bar** of the virtual screen by **double clicking with the left mouse button** in the background of the virtual screen.
Even if the title bar is hidden, you can still move the virtual screen by simply clicking into the background of the virtual screen and then dragging the window.
If you want to keep the virtual screen always on top of all other windows, open the system menu with the right mouse button and select the menu item **Stick it**.

**Save Wins DT in the context of the Virtual Desktop**
You can save all windows positions with the menu item **Save Wins DT** in the submenu **Window-Manager**.
This also works in the context of the Virtual Desktop. If you select this menu item, and the Virtual Desktop is on, you are asked if you want save the positions in the coordinates of the virtual desktop. If you here answer with NO, first the desktop will be **restored**. This means, all windows will be moved into the first virtual desktop, before saving theirs positions.

## context-sensitive Add-Ons for New Menus for Windows

If you press the right mousebutton over a windows element, which is listed below, you get a popup menu, so called context-sensitive menus.
The option **Context-Sensitive** in the General Properties dialog must be active to use the add-ons.
The menuitems are defined in the file ADDONS.MNU and can be changed by the user. Refer to "INI-file, Menuitems" and "Internal Commands" in the helpfile, how to customize this menuitems.

Edit controls (Notepad)

File-Manager

Dos-Box

Scrollbar

Listbox

### Text-Edit

The Menu for Text-Edit-controls (entry `Edit` in the section `[WinClasses]`) appears, when the mousecursor is over a normal windows text-edit-control (f.e. Notepad or the 'File' of an File - Open dialog).

- **Copy**: To copy the marked text in to the clipboard.
- **Insert**: Insert the text in the clipboard in the edit-control.
- **Cut**: To deletes the marked text and copy it in the clipboard
- **Undo**: To undo the last changes.
- **Search Text:**
  **Search >**: To search the highlighted Text to the end of the text.
  **< Search**: To search the highlighted Text to the beginning of the text.
  If you press the **Shift**-key while selecting **Search >** or **< Search** the case is relevant.
- **Save**: To save the Text of the edit-control in a file.
  The context sensitive Menu for edit controls can save files in more formats:
  TextFiles: The text will written without changes.
  With Layout: Normally if you use the 'line break' feature in Notepad (last menuitem of the menu 'Edit') and you save the text, the line breaks are lost. But if you use the Menuitem 'Save' in the context sensitive Menu, this soft line breaks are converted in hard line breaks. But if you want better layout functions in edit controls use the menuitems in the Submenu **Text Format**.
  Dos2Win: The text will saved converted from ASCII char set to ANSI char set.
  Win2Dos: The text will saved converted from ANSI char set to ASCII char set.
  UNIX: The text will saved converted from ANSI to ASCII char set. Also the linefeed will converted from DOS to UNIX convention.
  There is no regulation, which file extension you use (*.unx is just a proposal from me).
- **Text Format:**
  With this Menuitems you have a formatting program for simple texts. It can be used with notepad or any other multiline text control.
  The menuitems are only samples. You can create your own formats for your texts. If you highlight text, this will be interpreted as one paragraph. You haven't to highlight the whole text from the first char of the first line to the last char of the last line, because for the formatting the marked text will automatically expanded to complete lines.

```
- Text konvertieren:
Diese Menüpunkte manipulieren einen markierten Text:
Großbuchstaben: Der markierte Text wird in Großbuchstaben
umgewandelt.
Kleinbuchstaben: Der markierte Text wird in Kleinbuchstaben
umgewandelt.
DOS -> Win: Der markierte Text wird von DOS
ASCII-Zeichensatz in den Windows ANSI-Zeichensatz
konvertiert (insbesondere, wenn Sie anstatt der Umlaute nur
seltsame Zeichen sehen, ist diese Option ein Versuch wert).
Win -> DOS: Der markierte Text wird von Windows
ANSI-Zeichensatz in den DOS ASCII-Zeichensatz konvertiert.
```

In this sample the paragraph begins with "- Text ..." and reach down to "...konvertiert". (Yes, it's German ;)
If you use one of the formatting Menuitems, the text could showed like:

- Text konvertieren: Diese Menüpunkte manipulieren einen markierten Text: Großbuchstaben: Der markierte Text wird in Großbuchstaben umgewandelt. Kleinbuchstaben: Der markierte Text wird in Kleinbuchstaben umgewandelt. DOS -> Win: Der markierte Text wird von DOS ASCII-Zeichensatz in den Windows ANSI-Zeichensatz konvertiert (insbesondere, wenn Sie anstatt der Umlaute nur seltsame Zeichen sehen, ist diese Option ein Versuch wert). Win -> DOS: Der markierte Text wird von Windows ANSI-Zeichensatz in den DOS ASCII-Zeichensatz konvertiert.

The paragraph is now justified at the left and the right margins with a left borderspace of 5 chars, with a length of 65 char and the first line has an negative indentation of 2 chars.

1=0|Left 7, First -2, 65 Length, Justified|MNU_CALLDLL ADDONS.DLL EDIT_FORMAT @HWND 7 -2 65 3<1;ADDONS.DLL 25

Important are the settings between "@HWND" and "<1.

If you use a indentation in the first line, you have to subtract it from left borderspace to get the normal borderspace for the paragraph.

The meaning of the last values for the justify is:

- 0 = left
- 1 = right
- 2 = center
- 3 = left and right

Comments:

If you use 'center', the lines are not exactly in the center, if the chars of the line aren't divide by 2 without a rest.

You can simple assemble your own formatting by editing this INI-Lines.

- **Statistics**: With this menu item you can get some information: current line number out of total number of lines; length of the current line, length of the highlighted text. If you marked more than one line, the value for the length of the line may be wrong.
- **Upper, Lower**: With the menu items "Upper" and "Lower" the highlighted text is converted in upper or lower case respectively.

## File-Manager

- **Copy Files**: Copy all highlighted filenames into the clipboard. The filenames (including the path) are separated with a white space.
  Please be patient; to copy 550 filenames it takes a 486 processor @ 33 MHz 1 minute (is the segmented architecture of DOS the problem?!).
- **Edit File**: The highlighted File in the File-Manager is opened with the File Editor you have chosen in the dialog **General Properties**.
- **View File**: The highlighted File in the File-Manager is opened with the File Viewer. You can choose which viewer to use also in the dialog **General Properties**.
- **Print Files**: If you have highlighted a file in the File Manager you can print it with the menu item "Print File". If the file is a document (associated with an application) the corresponding program will print the file, otherwise the 'FileEditor' (declared in the section "FileViewer" of the dialog General Properties) will try to do it.

- **Copy Path**: Copy the Path (name) into the clipboard.

- **File Operation**: With the menu items in this Submenu, you can have access to the most common file operations. Refer to the ADDONS.MNU for more hints on how to customize this submenu.

### DOS-Box

The menu for **DOS-Boxes** (entry `tty` in the section `[WinClasses]`) appears if you press the right mouse button over a DOS-Box (You can run DOS-Sessions in a window if you are running Windows in enhanced mode).

- **Mark**: Switch a DOS-window from the Edit-Mode to the Mark-Mode. After using the Menu item **Mark**, you can mark text in the DOS-Box.
- **Copy**: To copy the marked region to the clipboard.
- **Insert**: 'Insert' the text from the clipboard in the DOS-Program - i.e. the keys corresponding to the text are sent to the DOS-program.

## Scrollbars

- **Top:** To move the scrollbar to the beginning (up or left)
- **Bottom**: To move the scrollbar to the end (bottom or right)

(This feature doesn't function with all Scrollbars)

### Listbox

- **Copy**: To copy the marked list item to the clipboard.
  (Be careful, some listboxes don't like it.)