MagicMenuE

COLLABORATORS

	TITLE : MagicMenuE		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY		February 6, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

MagicMenuE

1.1 Magic Menu Manual

```
     *******************************************************************

                    M A G I C   M E N U

                        V 1 . 2 9

     ******************************************************************


                © 1992 – 1993 Martin Korndörfer


               **************************
               *                        *
               *   I M P O R T A N T !   *
               *                        *
               **************************

     MAKE SURE YOU READ THE
               INSTALLATION-NOTES
               !!!

     IF YOU STILL USE MAGIC MENU 1.21 OR OLDER, YOU SHOULD READ THE

               RELEASE HISTORY
               !


                    C O N T E N T S
                    ===============


        1. Introduction


           1.1     Motivation
```

1.2 motivation

```
                    1.1 Motivation
                    **************
```

Developing AmigaOS 2.0, Commodore's developers have re-worked the outer
appearance of the Amiga's User Interface in an astonishing way.

... too bad they forgot Intuition's menus ...

As I wanted to flee from the dull menu life and the many programs that are
available did not satisfy me, I sat down at my keyboard and started to
type...

(Programming the Amiga means _lots_ of fun!)

1.3 features

```
                           1.2 Features
                  ************
```

Magic Menu includes the following features:

  o All Intuition menus are replaced by Magic Menu. Therefore, all programs
     using standard menus will appear in the new outfit.

  o Magic Menu supports
               Pull-Down-Menus
               as well as
               Pop-Up-Menus
               .

  o Menus are either displayed using the
               Standard Look
               , or the modern
      AmigaOS 2.0 style
               3D-Look
                o Menus may be controlled exclusively using the
               keyboard
               , there's no

need to grab the mouse anymore.

o   There are three different types of using the menus:

>   Standard
>   ,
>   Press Once
>   and
>   Select
>   .

o   Magic Menu can be
>   configured
>   in different ways, you may for example
   define handling and appearance of Pull-Down and Pop-Up menus seperately.

o   Magic Menu remembers every menu's last selected item, displaying Pop-Up
   menus the next time at a position allowing quick selection of the same
   or neighboring items.

o   if a menu is selected while the currently active screen is not visible,
   the screen is popped to the front for the menu display. After selecting
   an item, it is returned to the back.

o   if there is no input in a configurable time (moving the mouse, keypress)
   the menu selection is canceled, avoiding the machine to be blocked too
   long.

and lots more...

1.4   requirements

>   1.3 System Requirements
>   **********************

There is only one requirement for using Magic Menu: you have to be using
AmigaOS 2.04 (37.175) or higher.

Magic Menu will also work with Kickstart 3.0, it will even utilize special
features of the new operating system.

Additional libraries are not needed.

If you are still using Kickstart 1.3 or older, all you can enjoy is a
requester reminding you that your are STILL not up-to-date...

1.5 installwb

```
                           2.1 Installation from Workbench
        *******************************
```

Magic Menu only consists of one single program. The easiest way to install it is using the Workbench.

To install, just move the "MagicMenu"-icon to the "WBStartup"-drawer of your boot partition. That way Magic Menu is automatically started every time you boot your system.

In order not to open the
                 configuration-window
                  every time you start
MagicMenu, you should set the Tool Type CX_POPUP=NO in the WB information window of MagicMenu. Check your Amiga manual to learn how the information window works.

MagicMenu's configuration is saved in the program icon. You can change these settings directly using the Info-function of the Workbench. Anyway, using MagicMenu's configuration window is a lot easier to use.

1.6 installcli

```
                            2.2 Installation from CLI
         *************************
```

MagicMenu may also be run from the CLI. The settings will also be read directly from the icon, which has to be copied together with the program file.

MagicMenu detaches from CLI itself, you don't have to use RUN. Executing MagicMenu a second time will display the
                 configuration-window
             .

Command line parameters are not supported in the current version.

1.7   installnote

2.3 IMPORTANT NOTES
****

MagicMenu has to patch some OS-routines. This does not bring any problems,
as long as no other program patches the same functions.

It may happen that MagicMenu can't be removed from memory, when another
program installs itself. MagicMenu will recognize this and will give out an
appropriate message,

To avoid problems of this kind, you should use a program like "SaferPatches"
from Martin Adrian or similar. When using SaferPatches together with
MagicMenu, make sure you set the TOOL TYPE option
                SMARTPATCH
                to NO.
Otherwise, leave it set to YES.

References:

                Known Problems of MagicMenu

                Programmer Information

1.8   removing

2.4 Quitting Magic Menu
****

Magic Menu can be removed from the system in three different ways:

- Selecting QUIT in the
                configuration-window
                - Using AmigaOS 2.x's "Exchange" program found in one of your ←
                  system drawers

- Sending a CTRL-C to the MagicMenu process.

Please note that it might happen that MagicMenu can't be removed from the
system at once.

After sending a message to another process, MagicMenu will wait for a reply.

During that time, MagicMenu will not finish. After a certain time MagicMenu
will get a timeout for this message and will exit anyway.

A lot more difficult is the situation when a program that was started AFTER
MagicMenu patches the same system functions. MagicMenu will notice that and
display a message. Most of the time you can only select for MagicMenu to be
disabled, it will then stay in memory but won't do anything anymore. In some
rare cases you might want to select "Force uninstall". Make sure you know
what you are doing, because if you don't, the systems reliability will be
affected. See also:
                IMPORTANT NOTES
                and the Tool Type
                SMARTPATCH
                    .

1.9   confbasic

                                    3.1 General Configuration
                        ****

MagicMenu saves all setting as TOOL TYPES of its own icon. Of course you can
change them there directly, some can only be changed by this way. But most of
the settings can also be changed more comfortably by using the graphical user
interface.

To open the configuration window of MagicMenu, there are three possibilities:

– execute the program a second time

– using the program "Exchange" (to be found on your workbench disk in the
  directory Tools/Commodities):   Select the MagicMenu entry and then "Show
  Interface"

– the easiest way is by using MagicMenu's
                HotKey
                    . If you didn't change it,
  it's the combination of the three keys <CTRL>–<ALT>–<SPACE>.

The window is divided into three areas:

– Setup for Pull–Down–Menus

– Setup for

                Pop–Up–Menus
                – Setup for the
                keyboard control
                – General switches, modifying MagicMenu's reactions in all modes.

The window can be closed anytime using the Close-gadget in the upper left
corner or by selecting the button
                 HIDE
                 . In both cases, MagicMenu will
remain in memory and be active.

To save the current configuration, just select
                 SAVE
                 . All of the icon's
Tool Types will be set to the current setup.

To remove MagicMenu, select
                 QUIT
                 .

1.10 confpdbasic

                                  3.2 Pull Down Menus
                     ****

... is what any Amiga user is used to as standard menus. A line of menu
items in the top line of the menu, which open up when selected.

The pull-down menus of MagicMenu work the same way, except that you can
select between two different display types:

                 Standard-Look

                 3D-Look
               Plus, you can operate the menu using three different ←
                 methods (plus
 keyboard control):

                 Standard

                 Press Once

                 Select
               Which of these methods or display types should be used is set ←
                 using the left
area of the configuration window .

1.11 confpubasic

                                    3.3 Pop Up Menus
                          ****

You might need some time to get used to this option, But as soon as you do,
you'll probably find it being a very comfortable way of selecting menu items.

In   contrast to the normal
                 pull-down menus
                 the menu is not displayed at the
top of the screen, but rather as a box at any place where your mouse is.
You do not need to move the mouse to the top of screen anymore. Also, menus
are not displayed horizontally, but vertically.

Apart from that, pop-up menus work just like the normal pull-down menus.
They may be displayed using:

                 Standard-Look
                 und

                 3D-Look
                 and are controlled in one of these modes:

                 Standard

                 Press Once

                 Select
                Additionally, there is an extra setting called
                 CENTER BOXES
                 that will
probably make menu usage even easier.

Which of these display kinds and selection methods are used by MagicMenu is
defined using the center part of the configuration window .

1.12   confstandard

                         3.2.1 / 3.3.1 STANDARD menu selection
                    ****

Using the standard settings, MagicMenu will act just like the Amiga OS does:

After pressing the right mouse button, the menu strip (either pull-down or

pop-up, depending on the
                menutype settings
                and the position of your mouse)
will be displayed.

The menu strip will only be displayed as long as you keep the menu button
depressed. Moving the mouse pointer to a menu item will display its
subitems.

To select a particular item, move your mouse to it and release the right
mouse button.

To select several menu items (without leaving the menu), you can – just like
when using the normal Intuition menus – press the left mouse button when
pointing to an item.

TOOL TYPE:

    for
                Pull-Down Menus
              : PD_BUTTONMODE=STANDARD
    for
                Pop-Up Menus
              : PU_BUTTONMODE=STANDARD

Other methods of menu selection:

                Press Once

                Select

1.13 confpressonce

                              3.2.2 / 3.3.2 PRESS ONCE menu selection
                    ****

This option will probably be most appreciated by owners of trackballs, though
others might also find it useful: during menu selection, you do not have to
keep the right mouse button pressed, a short click will suffice.

After that, the menu strip is displayed (either pull-down or pop-up,
depending on the
                menutype settings
                and the position of your mouse) and you
can browse through the different menus and items just like in the

                    standard setting

                    .

To select an item, either the left or right mouse button has to be pressed.
If you use any of the buttons outside of the menu area, the selection is
aborted.

For multiple selects, use the left mouse button while holding the shift key.

TOOL TYPE:

    for

                Pull-Down Menus
                : PD_BUTTONMODE=PRESSONCE
    for

                Pop-Up Menus
                : PU_BUTTONMODE=PRESSONCE

Other methods of menu selection:

                Standard

                Select

1.14   confselect

                            3.2.3 / 3.3.3 SELECT menu selection
                    ****

... is the setting I prefer most. Especially for Pop-Up menus .

Just as with the
                PressOnce
                setting, the right mouse button only has to be
pressed once real short and the menu will be displayed (either as Pull-Down
or Pop-Up menu, depending on the setting of the
                menu type
).

In contrast to the other mode, the menu items aren't activated by moving the
mouse over them, they have to be selected using the left mouse button, too.
The same applies to subitems: to display them, their parent item has to be
selected.

The main advantage of this setting is that menus won't be displayed
accidently by moving the mouse a pixel too far. Some programs have very
complex menus, which almost cost some nerves finding a menu item. Using
SELECT you can use it like a switch box.

Selecting an item that has no subitems attached will return this item's
number to the appropriate program. To do a multiple select, press the shift
key while selecting an item.

TOOL TYPE:

      for

                   Pull-Down Menus
                 : PD_BUTTONMODE=SELECT
      for

                   Pop-Up Menus
                 : PU_BUTTONMODE=SELECT

Other methods of menu selection:

                   Standard

                   Press Once

1.15 conf3d-look

                                    3.2.4 / 3.3.4 3D-Look Display
                      ****

As of version 2.0 of the AmigaOS, Commodore has introduced a new 3D-Look in
the display of the user interface. MagicMenu implements this with menus.
This not only makes the menus look better, but usually also easier to use.

Additionally, there are two functional differences resulting from this
change:

- Intuition will only visibly select a menu item, if it is not disabled
  (displayed as ghosted entry). MagicMenu implements a similar behavior, but
  disabled items will have a kind of 'ghosted frame', mainly to make
  keyboard usage easier.

- Apart from menu items that directly result in some operation, there also
  some that turn an option on or off. These items are displayed with a small
  check mark when the option is turned on. Of course does MagicMenu support
  these, though the display has been polished up a bit. But there are also
  other menu items that have a check mark even though they react different

than the other items: mutually exclusive menu items. This could be used for
a color selection, where only one color can be used at a time. MagicMenu
will   display   these   just like OS 2.0's user interface does: it uses radio
buttons that either look recessed or raised, depending on whether the
item is selected or not. This way, these menu options can be recognized
easier as mutualy exclusive items.

In order to be able to have the 3D-Look, the screen displaying the data must
have been opened with at least 4 colors in a 2.0-compatible manner. This
might not be true for some older programs written for 1.3 or earlier.

Anyway, there are programs (like TagScreens from Martin Berndt) that modify
opened screens, so that they do get the 3D-Look. As sometime the colors will
not fit right anymore, MagicMenu's menus might not look three-dimensional
anymore. In such cases using the
                    standard look
                    might be preferable.

To switch between standard and 3D-Look, you can always press the Ctrl-key
while displaying a menu.

A few programs, mainly some that use menus for color selections, will not
work correctly with MagicMenu, check the
                    problems
                    section ...

TOOL TYPE:

    for

                    Pull-Down Menus
                  : PD_LOOK=3D
    for

                    Pop-Up Menus
                  : PU_LOOK=3D

1.16   confstandard-look

                              3.2.5 / 3.3.5 STANDARD-LOOK Display
                    ****

If you should not want the _three dimensional display_ of the menus, you may
always switch back to the old display style, while still enjoying all the
additional features of MagicMenu. While carefully implementing all of the
original flavor, there is only one slight difference to Intuition's menus:

When using the keyboard to control the menus, "disabled" items are
highlighted, too. This was necessary as doing without would make it hard to
find out where you actually are.

If a screen only has 2 colors, or has been opened using pre-2.0-methods, the
standard look display is used automatically.

TOOL TYPE:

    for

                 Pull-Down Menus
              : PD_LOOK=2D
    for

                 Pop-Up Menus
              : PU_LOOK=2D

1.17 confcenter

                                        3.3.6 CENTER BOXES
                         ****

This setting is only used with
                 pop-up menus
               , it doesn't affect the normal

                 pull-down menus
                .

If subitems are attached to an item, they will usually be displayed downwards
and to either the left or right side of the item; the programmer will have
taken care of placing the submenu (or GadTools for newer applications).

Using the setting CENTER BOXES, you instruct MagicMenu to forget about that
positioning and instead find a more efficient place for the submenu:

If no menu item has been selected before, the submenu will be vertically
centered relative to the menu item. Here's a little graphical demonstration
of that process:

      Without CENTER BOXES:

                     | Project |
                     | |
                     | Edit |
                     | ------- -----------
                     ||Options| Tabs |
                     | -------| |
                     | Texts | Pathes |
                      --------| |

```
                                 | Colors    |
                                 | -------- ----
                                 ||Height  | 10 |
                                 | --------|    |
                                 | Width   | 12 |
                                 |         |    |
                                 | Depth   | 15 |
                                  --------- ----
```

     With CENTER BOXES:

```
                                   ----------
                         --------| Tabs      |
                        | Project|           |
                        |        | Pathes    |
                        | Edit   |        ----
                        | -------| Colors  | 10 |
                        ||Options| --------|    |
                        | -------||Height  | 12 |
                        | Texts  | --------|    |
                         --------| Width   | 15 |
                                 |          ----
                                 | Depth    |
                                  ----------
```

When an item had been selected before, this item will be put directly beside
it's preceding item, e.g. after selecting 'Width' in the above menu, it would
be opened like this the next time:

```
                                   ----------
                                 | Tabs      |
                                 |           |
                                 | Pathes    |
                         --------|           |
                        | Project| Colors    |
                        |        |           |
                        | Edit   | Height    |
                        | -------| --------- |
                        ||Options||Width    ||
                        | -------| --------- |
                        | Texts  | Depth     |
                         -------- ----------
```

This way, selecting an item will be a lot easier the next time.

TOOL TYPE:

     for
                    Pop-Up Menus
                  : PU_CENTER=YES or PU_CENTER=NO

1.18   confkcbasic

                                            3.4 Keyboard Control
                            ****

One of the big advantages of MagicMenu is that you might as well put your
mouse aside, as menus can be used completely without mouse interaction, even
if it does not have a shortcut sequence (Amiga-key + another key).

If the menu is selected using the keyboard (either with the RIGHT ALT and
RIGHT AMIGA keys or with the combination set via
                START SEQUENCE
                ), the menu
strip will be displayed at the top border. (With keyboard control, only

                pull-down menus
                are used)

When first displaying a menu, one of the menus is selected. If no other menu
choice was done before, it will be the left-most menu, otherwise the one of
which an item was selected.   Use the cursor keys to select the different
menus.   Pressing Return selects a menu and displays it's menu items. With
items, either the submenus are displayed, or the current item is selected if
there are none.   The ESCape key lets you go up one level or leave the menu
selection without selecting an item.

The easiest way to select a menu or item is to press the first letter of a
selection.   If there is only one menu or item that fits to that letter, the
appropriate menu or item is selected, otherwise the cursor is only moved to
that item. In that case, pressing the letter again will let MagicMenu cycle
through the possible selections. Press the Return key to select the item.

For multiple selection, press the Return key together with the shift key. The
menu selection can be aborted anytime using the activation sequence (see
above).

As you may have noticed, MagicMenu tries to take a good guess at what you
want to do and remembers all the levels of the last selection.

There are the following settings available for keyboard control:

                ENABLE

                MOUSE TO BAR

                START SEQUENCE
                These settings are located at the right area of the
                configuration window

.

There are further descriptions of the keyboard sequences in the chapter

            keyboard layout
        .

1.19 confkcenable

                            3.4.1 ENABLE (Keyboard control)
            ****

If this option is turned on, the menus can be controlled via keyboard
commands.

For further information check the following chapters:

            Keyboard control

            Keyboard layout

1.20 confmousetobar

                            3.4.2 MOUSE TO BAR
            ****

Some programs (like DPaint or Directory Opus) only allow menu selection when
the mouse is somewhere in the menu strip of the screen, as the right mouse
button has another meaning at other areas.

As this would make menu selection via keyboard controls impossible, MagicMenu
will move the mouse to the top line when this option is turned on and the

            start sequence
            is deteced.

For further information, check the chapter
            Programmer's Information
        .

1.21   confkcstart

                                                3.4.3 START SEQUENCE
                                        ****

This input field allows you to select the key combination which activates the

                    keyboard-controlled menu selection
                .

All key names recognized by any other commodity may be used, see your Amiga
manual for further information on commodities. For your information, here's a
list of the names known to the Amiga OS Version 2.04. (Later versions know
some more)

        lshift   space
        rshift   backspace
        capslock   tab
        control   enter
        lalt   return
        ralt   esc
        lcommand   del
        rcommand   up
        numericpad   down
        shift   right
        alt   left
        caps   f1 - f10
                                help

For example, using "lcommand space" means that pressing the left Amiga-key
and the space key together will bring up the menu.

Entering an incorrect combination will result in an error message from
MagicMenu, the old setting will remain active.

The combination "ralt rcommand" (the right Alt and Amiga key) will always
work, independent of the start sequence entered (except if
                ENABLE
                is not
selected).

1.22   conftype

                                        3.5.1 TYPE (large Cycle Gadget)
                            ****

Use this switch between the different available menu types.

USE PULL DOWN MENU ONLY

    If this mode is selected, only
            Pull-Down menus
            will be used.

USE POP UP MENU ONLY

    This mode will result in only _Pop-Up menus_ being used, even if the
    mouse is in the title bar of the screen. Anyway, there is an
    exception: for keyboard control, Pull-Down menus are used.

USE PULL DOWN WHEN POINTER IN MENUBAR, POP UP OTHERWISE:

    The "automatic mode". When the mouse is somewhere in the title bar,
    Pull-Down menus are used, otherwise Pop-Up menus. This might come
    handy for people being used to moving the mouse to the title bar
    before pressing the right mouse button. Heart attacks and similar
    won't probably take place due to unexpected menu appearance.

TOOL TYPE:

```
for "USE PULL DOWN MENU ONLY":      MENUTYPE=PULLDOWN
for "USE POP UP MENU ONLY":         MENUTYPE=POPUP
for "USE PULL DOWN WHEN POINTER ...": MENUTYPE=AUTO
```


1.23 confmarksub

                  3.5.2 MARK SUB MENUS
                  ****

This option enables the marking of menu items that have a submenu attached.
When turned on, a small arrow will appear at the right side of such items.

If the text of this item is too long or if the program uses its own marks
(e.g. when using GadTools to layout the menus), MagicMenu's arrow is
disabled.

TOOL TYPE:

    MARKSUB=YES or MARKSUB=NO

1.24   confdborder

<div align="center">
3.5.3 DOUBLE BORDERS
****
</div>

Usually all menus of MagicMenu are drawn to appear as if they raise up from
the display, while selected items look like they are drawn recessed. Also,
the way the menu windows are drawn results in the effect of stacked
index-cards.

Using this option will change that appearance: Only the borders around the
menus have the 3d-effect, selected menu items appear raised. As the
difference is only optical, try out both kinds to find the one that suits
your personal taste best.

In
                Standard-Look
                this option does not have any effect.

TOOL TYPE:

    DOUBLEBORDER=YES or DOUBLEBORDER=NO

1.25   conftimeout

<div align="center">
3.5.4 MENU TIMEOUT
****
</div>

This function lets you select a timeout, after which a menu selection will be
automatically canceled if no mouse movement or key-presses occur. Values can
range between one and 120 seconds.

As MagicMenu locks the screen menus are displayed on (just like Intuition
does), this function will prevent the system from being locked.

Also, it may happen occasionally that MagicMenu und Intuition deadlock each
other (e.g. when moving a window while a menu is to be displayed, though
this should not happen...).   This deadlock-situation (2 or more programs
waiting for a resource the other programs have already locked) will be broken
up by the timeout of MagicMenu.

See also:

Possible Problems

Programmer Information
TOOL TYPE:

TIMEOUT=<Time in seconds>

1.26 confhide

3.5.5 HIDE
****

This gadget closes the
Configuration Window
, MagicMenu will remain acti-
vated. You may also click on the close gadget of the window.

1.27 confsave

3.5.6 SAVE
****

The current settings of MagicMenu are saved and will be available when
starting MagicMenu again the next time.

MagicMenu saves all its settings in the icon file from which it was started.
If MagicMenu was started from CLI, an icon file will be created.

1.28 confquit

3.5.7 QUIT
****

If this gadget is select, MagicMenu will attempt to remove itself from the
system. This might not succeed. Check the following chapters for more
information:

Quitting Magic Menu

Programmer Information

1.29   confsmartpatch

                                                       3.6.1 SMARTPATCH
                                     ****

Before removing itself from the system, Magic Menu checks all patches vectors
for other programs. If another program has patched a routine which has
already been patched by Magic Menu, the uninstallation will mostly fail.

There are some programs which are watching all vectors to help avoiding such
situations.

If such a program is running in your system, Magic Menu would always think
that its patches have been changed.

The error message may be ignored in this case, simply click on "force
uninstall". To turn off the security check of Magic Menu set the Tool Type
SMARTPATCH to NO.

TOOL TYPE:

     SMARTPATCH=YES or SMARTPATCH=NO (Default: YES)

For further information, check the chapter
                 Quitting Magic Menu
                  .

1.30   confcheckwb

                                                       3.6.2 CHECKWB
                                     ****

Kickstart 2.04 contains a bug which causes MagicMenu to recalculate the
workbench menus whenever an icon is selected or a window has been
(de)activated. This costs cpu time and is completely senseless.

Since KS 3.0 this bug has been fixed by Commodore.

To make Magic Menu usable with 2.04, a special check is implemented to avoid
unnecassary recalculations of the workbench menus.

This option should always be set to YES, unless there are unexpected problems
with the workbench menus. It has no effects to any other menus.

```
TOOL TYPE:

    CHECKWB=YES or CHECKWB=NO  (Default: YES)
```

```
For further information, check the chapter
                Programmer Information
                .
```

1.31 confuselayer

```
                         3.6.3 USELAYER
                       **************
```

Magic Menu renders all menus in so called "layers", that means that the
system makes sure that all graphics are clipped into the menu box, even if a
menu is not valid.

But this slows down the rendering of the menus.

Because of that, it is possible to turn of this behaviour with this tool
type. If set to NO, Magic Menu does not use the layers.library to render the
menus.

```
TOOL TYPE:

    USELAYER=YES or USELAYER=NO  (Default: YES)
```

1.32 confkcraltrcommand

```
                       3.6.4 KC_RALTRCOMMAND
                     ********************
```

Despite the keyboard sequence set in the configuration, Magic Menu uses the
right amiga key together with the right alt key to activate the menu
selection, just like the workbench itself.

With this Tool Type you can turn off this feature.

```
TOOL TYPE:

    KC_RALTRCOMMAND=YES or KC_RALTRCOMMAND=NO  (Default: YES)
```

1.33 confcxprio

```
                              3.6.5 CX_PRIORITY
                             ****************
```

This option sets the order in which all commodities are receiving their events in the handler chain.

In most cases, it should be set to 0.

TOOL TYPE:

    CX_PRIORITY=<Priorität> (Default: 0)

1.34 confcxpopup

```
                                        3.6.6 CX_POPUP
                             **************
```

If this option is set to YES, Magic Menu pops up its
                 configuration-window
              right after it has been started.

It should be set to NO if Magic Menu is always started while booting the machine (e.g. with the WBStartup drawer)

TOOL TYPE:

    CX_POPUP=YES or CX_POPUP=NO (Default: YES)

1.35 confcxpopkey

```
                                        3.6.7 CX_POPKEY
                             **************
```

This tool type sets the keyboard hotkey to pop up the Magic Menu's

                 configuration-window
              .

The default sequence is "control alt space", but it may be set to any other valid keyboard sequence.

TOOL TYPE:

    CX_POPKEY=<keyboard sequence>   (Default: control alt space)

1.36   confdontwait

                            3.6.8 DONOTWAIT
                            ****

This option MUST be set if Magic Menu is started from within the WBStartup
drawer, otherwise workbench would wait forever for Magic Menu to complete its
task.

TOOL TYPE:

    DONOTWAIT

1.37   conftoolpri

                            3.6.9 TOOLPRI
                            ****

Workbench gives every started process a task priority which can be set with
this Tool Type.

This should be 0 for most programs.

But Magic Menu should get a higher priority, to avoid cpu consuming programs
from slowing down its display.

A value of 5 is recommended, as the workbench itself is running at this
priority.

TOOL TYPE:

    TOOLPRI=<Priorität>   (Default: 0)

1.38   confstartpri

                            3.6.10 STARTPRI
                            ****

This Tool Type has effect on programs in the WBStartup drawer only.

It determines the order in which the programs should be started. In most cases this should be set to 0.

Programs which are modifying the workbench menus (e.g. ToolsDaemon) MUST be started before Magic Menu.

TOOL TYPE:

    STARTPRI=<Priorität> (Default: 0)

1.39 hotkeys

                                   4.1 Hotkeys
                         ****

Hotkeys are keyboard sequences which always can be used for various actions.

Magic Menu uses the following three hotkeys:

    <CTRL> <ALT> <SPACE>

        Opens the
                 configuration-window
              . This sequence may be modified with
        the CX_POPKEY Tool Type.

    <LEFT AMIGA> <SPACE>

        Activates the menu selection with the
                 keyboard control
              . Works only
        with the keyboard control activated. This hotkey may be modified with
        the
                 START SEQUENCE
                 option.

    <RIGHT ALT> <RIGHT AMIGA>

        Also activates the menu selection with the keyboard control. This
        hotkey has been implemented for compatibility with intuition. It
        cannot be modified, but may be turned of with the Tool Type

                 KC_RALTRCOMMAND
                 .

1.40 ctrlkeys

                                4.2.1 Keyboard controls for all modes
                   ****

The   following   keys   are   always   working   while   a   menu   is   displayed,
irrespective of the current
                  selection mode
              :

    <ESC>

        Cancels   the   menu   selection.   In the keyboard control mode only the
        last   submenu   will   be   closed. To quit the menu completely it may be
        necessary to press <ESC> up to three times then.

    <CTRL> (while the menu is displayed)

        Switches   the   display between the 3D-Look and the standard intuition
        look. May be used if the colors of a menu are not displayed
        correctly.

    <CTRL> (press and hold BEFORE activating the menu selection)

        If any menu does not work with Magic Menu, it can be temporarily
        deactivated, and the original intuition menus are used instead. Of
        course none of the advanced Magic Menu features can be used then.

             Hotkey
             to activate the keyboard control

        If one of both activation hotkeys is pressed, the menu selection will
        be canceled immediately.

For further information, check the chapter
             keyboard control
             .

1.41 kckeys

                                4.2.2 Keyboard control
                   ****

The   following   keys   are   working   if   the menu is activated with one of the

```
            keyboard hotkeys
            only.
```

<RETURN>

    Opens a (sub)menu or selects a menu item. If pressed together with
    the shift key more than one item can be selected. The last item must
    be selected without pressing the shift key.

<CURSOR RIGHT> and <CURSOR LEFT>

    Selects one of the main menu items. If pressed together with the
    shift key the box it moved to the right or leftmost item.

<CURSOR UP> and <CURSOR DOWN>

    Selects a (sub)menu item. If pressed together with the shift key the
    box is moved to the top or bottom item.

Alphanumeric key

    The next (sub)menu item starting with the corresponding character is
    selected. If this item is the only one starting with this letter, it
    will be activated, too.

For further information, check the chapter
         keyboard controls
     .

1.42 questions

                  5.1 Question & Answer
            ****

F: "If I activate the menu, everything freezes. Why?"

A: Magic Menu has to lock the screen on which the menu is displayed. Just
    like intuition itself. That means no program is allowed to make an
    output to that screen. All other screens are not locked with one
    exception: windows cannot be opened. Programs that want to open a
    window have to wait until the menu selection has been finished.

F: "My program is running on another screen, but nevertheless it is
    stopped. Why?"

A: Does the programs make any outputs to a CLI (which is located on the
    workbench in most cases)? This happens without magic menu, too. Another

possibility is that the program tries to open a window while the menu
is displayed.

F: "The menus of some programs have really strange colors. Why?"

A: If a screen is opened with the 2.0 "tags" then it is supplied with the
     correct color table. On all other screens (which are opened in a 1.3
     compatible way) Magic Menu uses the standard display (like intuition).
     Programs like "TagScreens" promote these old screens to the new 3D-Look
     ones, but sometimes the colors are not quite correct. You may always
     switch to the standard display with the CTRL key, if the menus are not
     usable.

F: "In color selection menus some of the colors are not displayed
     correctly."

A: Press the CTRL key while the menu is displayed.

References:

               Known Problems of MagicMenu

               Programmer Information

1.43   problems

                              5.2 Known Problems of Magic Menu
                     ****

Some problems are unavoidable. This is a list of all known problem, so nobody
should be surprised by them:

  o On all window operations except   OpenWindow(), OpenWindowTagList()
     and ActivateWindow() it is possible to create a deadlock situation if
     the menu is displayed. This deadlock will always be ended by the
     timeout.

  o To display the menus in the OS 2.0 3D-Look, the text and background
     pen are reversed. I tried to implement this as intelligent as possible,
     but there may be some menus with wrong colors.

References:

               Question & Answer

               Programmer Information

1.44 profi

5.3 Programmer's information
****

I've tried to make Magic Menu as compatible as possible, that means that it
behaves just like intuition wherever possible.

  o While displaying a menu the corresponding screen is locked. The menu is
     rendered in the background and is copied into the screen using the
     layers.library routine SwapBitsRastPortClipRect().

  o To avoid crashing memory as a result of a bug or an illegally built
     menu, all drawing operations are clipped using the layers library.

  o Before displaying a menu, Magic Menu sends a message to all windows with
     the IDCMP_MENUVERIFY flag set. The active window may cancel the menu
     selection by replying the message with MENUCANCEL.

  o Magic Menu sends menu events through the input.device handler chain
     using the IND_WRITEEVENT command. Intuition dispatches a respective
     message to the active window. Even programs that are receiving the menu
     selection via the console.device are supported.

  o IDCMP_MENUHELP is supported.

  o The functions OpenWindow(), OpenWindowTagList(), ClearMenuStrip(),
     SetMenuStrip(), ResetMenuStrip() and ActivateWindow() are patched. All
     calls to these routines are suspended until any menu selection has been
     finished to avoid deadlocks.

References:

Question & Answer

Known problems of Magic Menu

1.45 history

6.1 History
****

To learn about the changes of the many releases of Magic Menu, you should
read the following chapter.

Maybe you simply want to read something about the problems I had to solve to
get the whole thing working.

The english version of the release history is very short, I'm afraid. You'll
have to read the german version, to get further information.

> Version 1.29
>
> Version 1.27
>
> Version 1.22
>
> Planned features for future releases

1.46   hist129

```
Revision V1.29
--------------
changed on Tuesday, 23-Nov-93  22:33:00
    Following bugs are fixed:
    - The field ie_Qualifier of all messages and events created by
      MagicMenu was unset (= 0). Now the field is initialized with the
      value returned by PeekQualifier() of the input.device. After this
      fix every program that interprets this field with menu events should
      work now (e.g. ADoc, GenAm) (Reported by Denis Gounelle)

Revision V1.28
--------------
changed on Wednesday, 06-Okt-93  23:24:00  by  --- Unknown ---.   LogMessage :
    Following bugs are fixed:
    - Special bug-fix for Mr. SIERRA@FUTURE.zer.sub.org!
      The multual exclude field is now checked with 32 bits instead
      of 16 bits.
```

1.47   hist127

```
Revision V1.27
--------------
changed on Sunday, 01-Aug-93  16:27:00  by  --- Unknown ---.   LogMessage :
    Following bugs are fixed:
    - The shifting of the menus introduced in V 1.26 has been removed
      again, because of trouble with the look of some menus. Instead,
      Magic Menu uses smaller images if the standard ones don't fit.
    - On some programs using their own checkimage it could happen that
      some pixels of the menuitem were erased.

Revision V1.26
--------------
changed on Saturday, 31-Jul-93  20:45:00  by  Martin Korndörfer.   LogMessage :
    Following bugs are fixed:
    - Menu items with 0x0 pixels could crash MagicMenu. This was the
      reason for the DOpus 4.1x crash. BTW, such menu items are NOT
      valid!
    - A mx menu item may now have the flag MENUTOGGLE set.
```

    – MM could not display menus exceeding the right edge of the screen.
      This has been fixed, now Term 3.4 has its "Dial" menu back.
    – The checkmark is rendered AFTER the texts again. There was trouble
      with some menus overwriting the checkmarks otherwise.
    – With KS 3.x some menu texts were to close to the checkmark image,
      because the system now scales the images. MM now shifts all texts
      if necessary to get enough room for its images.

```
Revision V1.25
--------------
```
changed on Thursday, 24-Jul-93 19:51:00 by Martin Korndörfer.   LogMessage :
    Following bugs are fixed:
    – In release 1.24, windows WITHOUT the NewLookMenu flag were not
      been recognized properly.
    Additions:
    – WindowToFront() is patched now, too. Now there should be no
      problem with programs which pop a window to the front.
    – After completing a keyboard selection, the mouse is moved to its
      previous position. (Suggestion by Garry Glendown).
    – New Tool Types:
      o USELAYER (Default: YES):
        if set to YES, Magic Menu uses the layers.library to avoid
        trashing memory as a result of improper menus or a bug in
        Magic Menu itself. If you set this Tool Type to NO, the rendering
        of all menus will be about 25% faster.
      o KC_RALTRCOMMAND (Default: YES):
        To disable the keyboard sequence "rcommand ralt" for the
        keyboard menu selection, set this option to NO.

```
Revision V1.24
--------------
```
changed on Thursday, 22-Jul-93 22:50:00 by Martin Korndörfer.   LogMessage :
    Following bugs are fixed:
    – Some programs try to implement a 3.0-ish look for their menus
      manually. This has caused some problems with Magic Menu.
      (Reported by Garry Glendown)

```
Revision V1.23
--------------
```
changed on Tuesday, 18-Mai-93 22:09:51 by Martin Korndörfer.   LogMessage :
    Following bugs are fixed:
    – In standard look, some menu items have been deleted while playing with
      the menu.
    – The checkmark has been overwritten on some menus.

1.48 hist122

```
Revision V1.22
--------------
```
changed on Wednesday, 24-Feb-93 13:42:39 by Martin Korndörfer.   LogMessage :
    Following bugs are fixed:
    – The checkmarks were not been updated correctly while selecting more
      items in the standard-look.
    Changes:
    – If CHECKWB is set to YES, a similar patch is made for PageStream.

        (Requested by Martin Huttenloher)

Revision V1.21
----
changed on Tuesday, 16-Feb-93 18:00:21 by Martin Korndörfer.   LogMessage :
    Following bugs are fixed:
    - If the right mouse button was pressed on a menu-less window, some
      memory got lost. (Reported from Paul Miranda and Peter Elzner).
    - With the RMBTRAP flag set, Magic Menu generated two MENUDOWN messages.
    - The font for the menu strip was not always correct.
    - MagicMenu now handles requesters (and double-menu requesters, too)
      correctly. (Reported by Gerhard Schmidt).
    - Two enforcer hits removed.
    Changes:
    - Virtual screens are now moved to get the menu always visible while
      selecting with the keyboard.

1.49   future

                              6.2 Planned features
                              ****

Following features are planned for comming releases of Magic Menu:

    o Detached menus

    o Extra settings for every screen

    o A completly rewritten user interface with more options

    o Localisation

    o Extra configuration program

    Suggestions are welcome!

1.50   credits

                              7.1 Credits
                              ****

Magic Menu is written by

    Martin Korndörfer
    Pommernstr. 15
    D-86916 Kaufering

    E-Mail addresses: Internet: korndorf@informatik.tu-muenchen.de
                       Z-Netz : M.KORNDOERFER@NATHAN.ZER or
                                 SYSOP@NATHAN.ZER

```
Stefan Stuntz   ... for his MFR and the idea for the name ... ;-)
Garry Glendown  ... for his tips about SAS/C and the patching
```

```
Angel
```

```
... you're simply magic!
```

1.51 licence

```
                        7.2 DISCLAIMER
                        **************
```

```
                        No Warranty
                        -----------
```

```
THERE  IS  NO  WARRANTY  FOR  THE  PROGRAM,  TO  THE EXTENT PERMITTED BY
APPLICABLE  LAW.   EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
HOLDER  AND/OR  OTHER  PARTIES  PROVIDE  THE  PROGRAM  "AS  IS"  WITHOUT
WARRANTY  OF  ANY  KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT
LIMITED  TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR  PURPOSE.   THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE
OF  THE  PROGRAM  IS WITH YOU.  SHOULD THE PROGRAM PROVE DEFECTIVE,  YOU
ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
```

```
IN  NO  EVENT  UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL  ANY  COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY REDISTRIBUTE THE
PROGRAM  AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE  OR INABILITY TO USE THE PROGRAM  (INCLUDING BUT NOT LIMITED TO LOSS
OF  DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR
THIRD  PARTIES  OR  A  FAILURE  OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS),  EVEN  IF  SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.
```

```
Magic Menu © 1992/93 Martin Korndörfer
```

```
Magic Menu is *NOT* in the Public Domain!
```

```
It may be distributed only if the following files are included:
```

```
    MagicMenu
    MagicMenu.info
    Deutsch/MagicMenu.guide
    Deutsch/MagicMenu.guide.info
    Deutsch/Benchmark
    Deutsch/Benchmark.info
```

```
    Deutsch/History
    Deutsch/History.info
    English/MagicMenu.guide
    English/MagicMenu.guide.info
    English/Benchmark
    English/Benchmark.info
    English/History
    English/History.info
```

As long none of the above files is missing, Magic Menu may be freely distributed on disk or on any network. A disk containing Magic Menu may not be sold for more than 5$ or an equivalent value.