PopupMenu

COLLABORATORS

	TITLE : PopupMenu		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		February 14, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

PopupMenu

1.1 PopupMenu Library V9.00

```
For more information, visit the homepage:

 www.boing.nu/pm
 popupmenu.amiga.tm  (mirror)
```

1.2 What is this??

```
Introduction

This is The One and Only Popup Menu Library you use when you want a nice
popup menu in your programs!
The included examples are compiled using SAS/C 6.58, they should work with
```

other compilers. If they don’t, let me know.

Features

* User configurable.

* Support for pulldown menus.

* Unlimited number of submenus.

* Unlimited number of mutually excluded items.

* Supports MultiSelect.

* Supports images and icons.

* MagicMenu images by Mario Cattaneo.

* Bold, italic, underlined and shadowed text.

* Coloured text.

1.3 Requirements

PopupMenu.library requires the following:

* AmigaOS 3.0 or higher.

Recommended:

* AmigaOS 3.5

* Picasso96 or CyberGfx

1.4 Author

Author of popupmenu.library, C developement files and examples:

Henrik Isaksson <henrik@boing.nu>

Author of E developement files and examples:

Daniel Kasmeroglu <raptor@cs.tu-berlin.de>

1.5 DISCLAIMER

COPYRIGHT
=========

This software is copyrighted by its developer(s). That means that you are NOT ALLOWED to modify the program(s) and documentation in any way. Especially you MUST NOT REMOVE the documentation or this text file.

You are NOT allowed to use this software or any part of it for any other purpose than that one mentioned in its documentation, this also includes any fonts, images or samples. If the developer(s) did NOT include the source code of the program(s) in this package you are NOT allowed to decompile any part of it.

DISTRIBUTION
============

This package is freely distributable. That means you are allowed to redistribute this package as long as you follow these points:

* You may NOT ADD any files to the archive!
* You may NOT CHANGE any files in this archive!
* You may NOT REMOVE any of the files in this archive!
* You may include the library file, the documentation and the preferences editor in your own distributions.

This package may be freely distributed via BBSs, InterNet/UseNet, software libraries such as Fred Fish's and Aminet® CD-ROM, and other similar electronic channels.

Disk magazines and services that charge extra for file transfers may NOT distribute it without written permission by the developer(s)!

DISCLAIMER
==========

By using this product, you accept the FULL responsibility for any damage or loss that might occur through its use or the inability to use it. The developer(s) of the software and the author and the translators of this "Copyright Note" can NOT be held responsible.

Some names used in this text are trademarks or registered trademarks. The use of these names does not imply that they are free.

1.6 Important things to think about when you're using popupmenu.library!

Distribution

* As stated in the license agreement, popupmenu.library is free for use in all applications, freeware, shareware or commercial.

* However, if your application is shareware or commercial, you should send either a full version, or a donation to the developer. Mail popupmenu@boing.nu for further information.
  This is volountary, but please consider supporting the developement of new versions this way. Sending a copy of your application costs you nothing, and it will not be spread to anyone.

* The copyright of configuration files created by you belongs to you, and

thus spreading them is up to you.

Technical

* If you want to use keyboard shortcuts in your application, you have to use the PM_FilterIMsg() function.
  (Unless you plan to handle the keyboard events yourself.)

* If you want the menus to be able to handle multiple selections, you have to use the PM_MenuHandler tag.

  NOTE: If your menu is opened with the left mouse button, you will not be able to select multiple items by clicking the right button, while you're holding down the left. To get around this problem, you must use the PM_UseLMB tag.

* If you want to follow Mario 'padrino' Cattaneo's MagicMenu style conventions, here are his rules:

  All contextual menus should have a title describing it's context.
  ("Trashcan" for a trashcan icon in a WB replacement software)

  The title should be printed using shadowed, centered text, and it should be white. (use the tags PM_Centered, PM_Shadowed and PM_ShinePen)
  The simplest way to do this is to use the PMMenuTitle macro.

1.7 History

The history of popupmenu.library:

9.00 Added PM_AbortHook(), a function that should be called periodically from within submenu constructor/destructor hooks, to find out if the user does not want to wait for the menu to complete.

Added PM_LocaleHook and PM_StringID which simplifies localization of the menus.

Added PM_GetVersion(), which simply returns the version number of popupmenu.library. This function is intended for use by the prefs editor.

Added PM_ReloadPrefs(), which forces the library to load the prefs file. This function is intended for use by the prefs editor.

Added new tags for advanced layout. (PM_Members, PM_LayoutMode)

Submenus now 'walk' across the screen in a different way. When you reach the right side of the screen, they will start popping up to the left of the parent menu, until you reach the left side again.

Added maximum depth for the shadows.

Changed the PopupMenu structure. It is now sized dynamically, simple items (eg. text-only items) require less than half as much memory as before.

"Old Style" separator bars used to be drawn with the wrong pen, now they're drawn with shadowpen.

"Old Style/Intuition" borders where incorrectly laid out as being 1 pixel wide, while their actual with is 2 pixels. This caused them to be overwritten sometimes.

Fixed bug with ColourBoxes appearing under AmigaKey images in check items.

Fixed keyboard shortcuts that didn't work with check items.

Added PM_HintBox, which makes the menu disappear when the mouse is moved. Intended for showing short hints when the mousepointer stops over a GUI element. Also known as "ToolTips".

8.20 Adjusted the MagicMenu style borders a little bit.

Added "Raised" mode for selected items.

Added E modules and examples by Daniel Kasmeroglu to the distribution.

Added "Intermediate spacing" settings for the space between images and text.

Added two new text modes, "Outlined" and "Embossed".

Added colour settings for text outline and shadow.

Submenus now behave a little better. They used to close at the same point as they opened on, now they close two pixels above this point.

Added settings for shadow intensity.

Added ability to adjust the vertical position of the item's text.

Fixed a bug which caused the non-window mode to trash the screen sometimes.

Fixed problems caused by omitting the PM_Menu tag.

Rewrote the menu list scanning routines in assembler. This saved 72 bytes, and hopefully made things a little bit faster.

Items with sub menus are now also selectable, if PM_NoSelect is set to FALSE. (not reversable)

Added support for $POO by Gareth Murfin.

8.00 Fixed a crash in the BigMenu demo.

Reworked the layout code and added settings for spacing between borders and text.

Fixed the ’dropbox’ border style mode.

Added text style settings.

Added intuition style menu borders.

Corrected the wide (title) separators in button border mode.

Added an option to turn off ’real’ shadows.

Added pulldown menu ’windowbar’ mode. (Menus show up at the top of the window instead of the screen bar.)

Replaced the icons with new NewIcons by Brian Gilbert.

Added two new functions, PM_InsertMenuItemA and PM_RemoveMenuItem.

Added a new tag, PM_Hidden.

Ghosted items now has MagicMenu-like ghosting, when MM2 borders and images are selected. Currently only the text will be affected.

Done a few minor visual adjustments.

7.50 Improved user input routines. The menus used to flicker sometimes when the mouse was moved.

New remap routine. Optimized by Ralph Wermke.

Added a submenu destructor hook.
PM_SubDestruct

7.40 Added a callback hook for dynamic allocation of menus:
PM_SubConstruct

Translated the remapping code to assembler.

7.35 Bugfix. BYTE read from $0 when a NULL pointer was passed to PM_Title.

7.3 Added a few new tags: PM_Right, PM_Bottom, PM_MinWidth, PM_MinHeight, PM_CenterScreen, PM_UseLMB.
See PM_OpenPopupMenuA for descriptions.

Removed a few Enforcer hits.

Made a few changes to the non-window mode. It should work better with CGFX now.

7.2 Added "Sticky mouse button" mode.

Removed the "Open menu when button is pressed/released" setting. (it wasn’t supported by any applications) Therefore the PM_Code tag is no longer needed.

Fixed one possible enforcer hit.

```
7.1 Bugfix. Sometimes the menu could show up on the wrong place.

7.0 The PM_FilterIMsg() function has been moved to a new library offset.
  This was done because the argument passing used the wrong registers.
  To provide backwards compatibillity, the old function is still left.

  The Image system is entirely rewritten. All used images are now
  stored in the configuration file. This saves some memory and will,
  in future releases, let the user load his/her own images.
  Image drawing should also be much faster, especially on chunky
  screens and on machines with chunky-to-planar converter.
  There may be some problems with prefsfiles from release 6.0.

  Fixed a bug in the separator bars. (The MM2 style was used even
  without MagicMenu2 borders selected.)

  Added the PM_ColourBox tag.

6.0 Made some changes to the API, for keyboard shortcuts and
  multiselect support. The library should still be 99% backwards
  compatible. Hacks like the previous version of the BigMenu demo
  will crash.

  Added keyboard shortcut support through PM_FilterIMsg().

  PM_ExLstA() added for easier listing of mutually exclusive item ID's.

  Added support for multiselect.

  Added menu border settings.

  Fixed the "standard 2d images".

  Fixed a problem with pulldown menus when "compact" was not
  selected.

  Moved some of the images out of the library, and in to the prefs
  editor. The needed images will then be stored in the prefsfile.
  (wich means that only the needed images will be in memory)

5.4 Moved the submenus 5 pixels up.

  Improved the deadlock detection.

  Improved the look of separator bars.

5.35  Optimized the pen allocation a bit.

5.3 Moved the submenus a bit to the left. Now it looks more like the
  other menus, and it's easier to open the submenus.

  Fixed a bug in pen allocation/freeing.

  Made a few adjustments to the separator bars.

  Fixed bug in "Submenu Delay" code.
  Fixed an enforcer hit in window mode.

  Got rid of a few unnecessary bytes again. (about 1k smaller)

5.2 Fixed delay bug in non-window mode.

  Improved the remapping of magic images.
  Right look and a few hundred percent faster. :)

  Pulldown menus should now work in non-window mode too.

  Fixed the shadows. They were 1 pixel to wide before.

  Added a new tag, PM_Shadowed, to make the text shadowed.

  Changed the PMMenu macro so the titles will use shadowed text.

  Fixed refresh problem in window-mode.

  Made the separator bars look more like MM2.

  And this time the library is actually 1272 bytes smaller!

5.1 Fixed the deadlock bug in non-window mode.

  Added MagicMenu2 Images, and it’s special remapping.
  (relative color values)

  Added support for pulldown menus.
  (works only in window-mode at the moment)

5.0 Opening the menu does no longer deactivate the active window.

  The menus can now use the blitter instead of windows, wich is much
  faster. (this has to be activated in the prefs-editor.)

  The menu font is now fetched from the DrawInfo, and not the menu’s
  parent window.

  There is now a space between the bottom of the text and the bottom
  of the select bar.

  Clicking the mousebutton very fast should no longer result in a
  menu that doesn’t dissapear. (as it should, when you release the
  button)

  Some of the tags send to OpenPopupMenu() has become obsolete.
  (The preferences has taken over)

  The input method has changed, so the library now requires
  commodities.library (wich is a disk library).

  The menu shadows now look like MagicMenu2 shadows.
  (the size increases for each submenu opened)

  Starting with this version, kickstart 3.0 is required. At least for
  the non-window mode. In window mode it _may_ still work with 2.0,
  but it’s not tested. So if you have ks 2.0, I would appreciate if
  you could try it out.

  Fixed a bug. Submenus could appear at the bottom of the screen
  when the mouse was moved quickly over an item.

  Added a new function, PM_AlterState().

4.3 Fixed a bug. (read from adress 0)

4.2 Now the menus can have shadows.

4.1 Images are now remapped correctly.

  Small changes for colour and image prefs.

4.0 More bugs fixed.

  A new demo, MenuVerify, shows how to use popupmenu.library with
  IDCMP_MENUVERIFY.

3.6 Bug fix. (never released, i got more bug reports...)

3.5 Now checks for the file ENV:PopupMenu.cfg, and if it exists,
  loads it, and replaces the default settings.

  Bug in PM_SubMenuTimer fixed.

  PM_Code added.
                Read
                 this!

3.0 Added PM_GetItemAttrs(), PM_SetItemAttrs(), PM_IsChecked
  and PM_FindItem.

  Added the tags PM_Left and PM_Top.

  Added a few more demos.

  Changed the naming of functions and macros to avoid interfering
  with other libraries.

2.0 Entirely rewritten and is now a shared library!

1.3 Added submenu support

1.2 Lots of new flags, and checkable menuitems

1.1 OpenPopupMenuPos()

1.0 First release
```

1.8 Future

The future:

- Asynchronous rendering into the menu through a user defined process under control of the popupmenu.library. This will provide support for cool things such as animations and icons that show up as they are loaded.

- Even more control over the layout of items through tables.

- Centering checkmarks in the icon column.

- Bitmap + mask support for the icons.

- Handling of menus that are too large to fit on the screen.

- Setting to open submenus only if the mousepointer touches an area close to where the submenu will open. (i.e. somwhere close to the submenu arrow.)

- Make all the graphics functions available in the library. (border drawing, 'blitter windows' etc.)

- BOOPSI menu item class.

- Datatype loading of images.

- Font settings. (maybe)

- Mail me if you have any ideas!

1.9 "

Special thanks goes to:
(in order of apperance)

* Stefan Sommerfeld for his nice bugreports, help with the non-window mode, and the remapping code (too bad I coudn't use it :( ).

* Trond Werner Hansen for his shadow-drawing code!

* Mario 'padrino' Cattaneo for all the MagicMenu2 images and the other MM2 specs. BTW, if you don't want to upset him, remember to spell padrino with a small p... ;)

* Olaf 'Olsen' Barthel for his useful hint about deadlock detection.

* Mattias P. Eriksson for his Enforcer hits

* Alfred P. Schwarz for his Enforcer hits

* Sebastian Bauer for even more Enforcer hits ;)

* Ralph Wermke for the optimized remap code

```
* Brian Gilbert for the icons

* Georges Goncalves for being an "endless suggestion list generator" :)

* Daniel Kasmeroglu for the E modules and sources

* Martin McKenzie for information about MenuVerify
```

1.10 Macros Example

```
struct PopupMenu *menu;

/*
 Creating a menu with the macros...
*/

menu = PMMenu("The title of the menu"),
 End;

/*
 Creating a menu like the one above, but with items...
*/

menu = PMMenu("The menu title"),
   PMItem("The item text"),
     /* Tags for the item */
   End,
   PMItem("The item text (2)"),
     /* Tags for the item */
   End,
 End;

/*
 Creating a sub-menu for an item...
*/

menu = PMMenu("The menu title"),
   PMItem("Item with submenu"),
     PMSimpleSub,
       /* Here are the items... */
     End,
   End,
 End;

/*
 Creating an exclude ID list for PMExclude can be done in two ways.
 This is the first:
*/

 PM_Exclude,
   PMExl ExID(1),
     ExID(2),
 End,

/*
```

```
  This is the second way:
 */

  PM_Exclude, PM_ExLst(1, 2, 3, 0), End,
```