

NListtree.mcc

Carsten Scholling

COLLABORATORS

	<i>TITLE :</i> NListree.mcc		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Carsten Scholling	August 25, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	NListtree.mcc	1
1.1	NListtree.mcc v18.7	1
1.2	Disclaimer	1
1.3	Introduction	2
1.4	Licence	3
1.5	What do you need to run this?	3
1.6	Installation	3
1.7	How to use this thing?	3
1.8	Known bugs	4
1.9	Credits	5
1.10	History	5
1.11	Planings for next releases	8
1.12	Where to get the updates?	9
1.13	Acknowledgements	9
1.14	MagicUserInterface	9
1.15	NList.mcc, NListview.mcc, NFloattext.mcc, NListviews.mcp	10
1.16	mcc-install	11

Chapter 1

NListtree.mcc

1.1 NListtree.mcc v18.7

NListtree MUI Custom Class

©2001 Carsten Scholling

Disclaimer	- Boring stuff...
Introduction	- What is it?
Licence	- Pay? Why and where?
Requirements	- What is needed?
Installation	- Easy doing...
Usage	- How to use this thing?
Known bugs	- Read THIS before sending bug-report!
Author	- Who did it?
History	- Small, but fine.
To do	- What is to be done in the future?
Updated	- Where to get the updates?
Credits	- Brablabla :-)
About MUI	- MUI copyright notice
About NList	- NList copyright notice
About MCC-Install	- MCC-Install copyright notice

1.2 Disclaimer

DISCLAIMER

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.

1.3 Introduction

Introduction

This MUI custom class provides you with an easy way of displaying list entries in a sorted form, a so called tree-structure. A Tree-structure will be selected, when data should be optical seperated from each other.

Today i know of several list tree classes, but none of them fullfilled my needs:

First, there is the old Listtree.mcc from Klaus Melchior which is used in several applications. For this one, there is no development since many month (year?). This class works stable, as long as the programmer knows about the several bugs.

Then a class called NListtree. Only a few bytes are different from Listtree. Mainly the superclass strings are patched (List.mcc -> NList.mcc) to use NList instead of List.

NewListTree from Ole Friis which is unchanges in alpha state (version 0.2) since january 1999.

And, the last, TreeList from John Haubrich. He has dropped the project and his Amiga somewhere between the 10. or 11. of December 1999.

The sourcecodes of both last named classes are public. Today i do not know about someone who has worked on these codes.

NListtree is intresting for programmers and the end-user also and easy to use. So far as implemented by the programmer and needfull in the application environment, this class allows the fully free recreation of the list using drag&drop (short d&d) with support of multiselection.

First time in a list tree class it is possible to move complete structures. That means, that the complete tree of the moved entry, back to the root-list, is recreated at the destination if the programmer allows this. The same for copying entries, which is also implemented.

1.4 Licence

Licence

This class is freeware for end users, but polymorphware for developers. That means that anyone who makes a shareware, giftware or commercial application using NListtree must send me the same fee he expects for his program.

If you have written a shareware program using NListtree and you expect \$20 for your hard work, i expect \$20 from you for my work. Easy, isn't it?

To make the things easier: If i am interested in your application (please ask before), we can make a deal. You are sending me a registered and full working release of your application including updates. Then there is no other fee (and will not ever).

I'm interested in who is using my classes in his own (free) programs. So don't hesitate to contact me. (just an e-mail telling what you're doing with NListtree :)

1.5 What do you need to run this?

Requirements

- * MUI 3.8 (not tested with earlier versions)
- * NList.mcc 0.90 (NList.mcc 19.98)

1.6 Installation

Installation

Just use the supplied installer script or copy the classes NListtree.mcc and NListtree.mcp to the MUI:libs/MUI directory and the supplied images to MUI:Images/aphaso on your harddisk.

1.7 How to use this thing?

Usage

NListtree is almost the same and reacts the same as Listtree does. But the preferences program has some differences:

In Listtree configuring Windows 95 style causes Listtree to draw the open/closed sign directly. NListtree uses the user selectable images for these signs too. Style Win98+ has an additional image for the drawer to make it really look like Windows95/98.

Some new imagery especially for the Windows styles is distributed with NListtree. They can be found in the directory Images/aphaso.

The user can control the list tree by the MUI keys, this means a node is opened with "Right" and closed with "Left". "Up" and "Down" are the same as in normal List class. Check your MUI NList prefs for the specified keys.

Simple clicking on a node sign or double clicking somewhere else in the tree column and/or other columns (this will be defined by the programmer and should be stated in the application docs) toggles its status from open to closed and vice versa.

Multiselection is implemented and reacts as configured in the NList preferences or like programmers choice. This should be written down in the application docs. Please note, that multiselection and drag & drop in a tree with more than one selected entry can be confusing sometimes:

- in multiselection mode, you have to move the mouse pointer out of the list while holding down the left mouse button, to start dragging entries around
- the default key (qualifier) for dragging in multiselect lists (NList.mcc) is CTRL. Pressing CTRL causes NListtree (NList) to immediately start drag & drop

So please check out the demo program NListtree-Demo on how to use this feature by dragging some entries around. But first read the next section.

Drag&Drop capabilities:

If possible or needfull for the situation NListtree is used by the programmer, you can drag & drop entries. This means you can drag one entry and drop it on the same listtree again. While dragging an indicator shows where to drop.

Drag a	Drop on	Result
leaf	leaf	Exchange leaves.
node	leaf	Nothing happens.
entry	closed node	Moves entry into the list.
entry	open node	Moves entry to defined position.

You can not drop an entry on itself, nor you can drop an opened node on any of its members.

1.8 Known bugs

Known bugs

- auto unselection does not work in all cases

1.9 Credits

Author

Carsten Scholling
Grosser Lahweg 34
31020 Salzhemmendorf

Germany

e-mail: cs@aphaso.de
WWW: <http://www.aphaso.de>

Please write me suggestions, bugs, wishes...

1.10 History

History:

V18.7 - Public release

- ADDED: Now, ...Select_Active with ...Select_Ask returns the active entry instead of 0.
- FIXED: A selected entry in a list node does not memory its state over 2 or more close/open on this list node (Thx to Andrija Antonijevic)
- FIXED: ...Select_Active ...Select_Toggle caused an alert without active entry (Thx to Andrija Antonijevic)
- FIXED: After a MUJA_NListtree_Quit, sometimes the tree was refreshed twice. This is a very old bug i've forgotten for some weeks. (Thx to Sebastian Bauer for reporting it again)
- ADDED: MUIM_NListtree_Clear
- ADDED: Check for outdated NList.mcc. If version is lower than 19.97, NList aborts.
- REWR: Rewritten selection code for new NList feature. Attention! With a lower NList version than 19.98, NListtree disables his multi-selection feature now!

V18.6 - Public Beta release 5

- FIXED: Fixed bug introduced with multiselection that allows dragging of entries to their children what ends up in a crash (reported by Sebastian Bauer, thx)
 - FIXED: Rewritten entry position calculation code. Never more accidentally deleted entries!
 - FIXED: Drop type "Sorted" never occurred. Fixed!
 - FIXED: Renaming the user field was not possible. Fixed.
 - ADDED: Added MUIV_NListtree_Insert_ListNode_ActiveFallback (see autodocs for more details) (suggested by Sebastian Bauer)
 - ADDED: Added a second find method and the corresponding hook for user data. This is meant as a backup if searching for different fields
-

- and/or values (MUIA_NListtree_FindUserDataHook, MUIM_NListtree_FindUserData).
- ADDED: Added MUIA_NListtree_ShowTree for enabling/disabling the tree. This is usefull for lists which can be sorted in different cols.
- ADDED: Added spanish catalog (Thx to Javier de las Rivas).
- ADDED: Added trigger attribute MUIA_NListtree_SelectChange.
- FIXED: Inserting open nodes into a closed node caused NListtree to miscalculate the positions (reported by Luca 'Hexaae' Longone, thx).

V18.6 - Beta release 4

On the fly-release for some persons.

V18.6 - Beta release 3

- Added MUIA_NListtree_DropType, MUIA_NListtree_DropTarget and MUIA_NListtree_DropTargetPos.
- Now, MinLineHeight will only be set when image is heigher than the current minimum.
- When using Next/PrevSelected method and deleting selected entries, machine crashes. Fixed.
- Added MUIV_NListtree_FindName_Flag_Reverse (suggested by Thomas Igracki)
- Added MUIV_NListtree_FindName_Flag_StartNode (suggested by Thomas Igracki)
- Added italian catalog. Thanks to Andrea Zanellato for the translation.
- Added NULL pointer check to internal construct/destruct hooks.
- Added MUIV_NListtree_Remove_TreeNode_Selected.

V18.6 - Beta release 2

- Changed look and feel of the mcp.
- mcc was not closed in mcp. Fixed!
- Source cleanup.
- Modifications to the tree drawing functions. Should look a little bit nicer now.
- Added MUIA_NListtree_CopyToClipHook and MUIM_NListtree_CopyToClip. Suggested by Harald Frank.
- Added an own context menu for copying strings to clipboard units.
- Added a translation package for easy translation of NListtree catalogs.

V18.6 - Beta release 1

- MUIV_NListtree_MultiSelect_Shifted now works as expected
 - Crashes/hits on exit (dispose) finally removed!
 - Sorting a closed node forced entries to show up. Fixed! (Thx to Thomas Igracki)
 - Reimplemented (oups!) MUIV_NListtree_GetEntry_Flag_SameLevel and MUIV_NListtree_GetEntry_Flag_Visible (Thx to Thomas Igracki)
 - DoubleClick again: Removed silly bug that causes double notification on double click. Hey Matthias: Remove workaround ;-)
-

- Added some ball images from Henning Thielmann, thx a lot.
- Fixed DropType bug which caused an eight bit left shift of the original values.
- Special image in .mcp was not ghosted when starting MUI without a saved configuration. Fixed. Bumped to V18.6.
- Without configuration, no or a wrong image was shown in tree. Fixed! Reported by Harald Frank, thx.

V18.5 - Public release

- Added preparse string pointer to MUIP_NListtree_DisplayMessage (suggested by Andrija Antonijevic)
- Made drop types public (same values as NList drop types) (suggested by Sebastian Bauer)
- MUIA_NListtree_Doubleclick is now triggered if return key is pressed (reported by Sebastian Bauer)
- NULL pointer will be replaced with "*** NULL POINTER ***" text now.
- Multi test hook will now be launched in all cases when multi selecting (reported by Thomas Igracki)
- Removed bug in NextSelected function which caused returning of trash (reported by Thomas Igracki, thx).
- Added MUIM_NListtree_PrevSelected (suggested by Thomas Igracki)
- Added MUIV_NListtree_FindName_Flag_Selected to only find selected entries in MUIM_NListtree_FindName (suggested by Thomas Igracki)
- Fixed bug in DoubleClick: Setting DoubleClick to any other than All has no effect (reported by Mark Harman, thx)
- Added new DoubleClick value: MUIV_NListtree_DoubleClick_NoTrigger. This means, that double clicking an entry is disabled including disabled trigger (attribute and method).

V18.4 - Broken public release, sorry :(

- Added MUIM_NListtree_InsertStruct
- Added some new internal attributes and methods ;-)
- Fixed autodocs a litte bit (reported by Allan Odgaard)
- Added nesting count to MUIA_NListtree_Quiet (suggested by Allan Odgaard)
- Fixed Enforcer hit when deleting the last list entry in root node (reported by Matthias Böcker)
- Fixed Enforcer hits when listview is the active object, no entry is active and using cursor left/right
- Added MUIM_NListtree_Active and MUIM_NListtree_DoubleClick methods.
- Removed PoolWatch-hits (reported by Andrija Antonijevic. Thx!) and written the book "Do Not Assume That Making Only Two Small Changes In A Function Does Not Affect The Functionality"
- Removed (hopefully) crash on exit (dispose)

V18.3 - Public release

- MUA_NListtree_Quiet does not work as expected (reported by Thomas Igracki, Thx!)
 - User defined compare- and findname-hooks crash! Fixed! (reported by Thomas Igracki, Thx!)
-

- Now, NListtree is really quiet if MUIA_NListtree_Quiet is set!
- Reactivation of copied, moved and exchanged entries works better now.
- OpenAutoScroll now works as expected, added new config item Open scroll (Thx to Matthias Bethke).

V18.2 - Public release

- Added multiselection feature
- Fixed some small cosmetical problems
- Implemented arrows as MUI images
- Implemented Copy method
- Implemented KeepStructure flag for recreation of the complete structure in Move and Copy

V18.1 - Internal release

- Completely rewritten graphics stuff. Now in all cases predefined and precreated images where used. This fixes many graphic bugs occurred in the old 18.0.
- After deleting an entry, the old entry remains internally active. Using this entry crashed the machine. Fixed!
- Moving marker with the cursor keys did not trigger MUIA_NListtree_Active. Fixed!
- When deleting entries, MUIA_NListtree_Active was triggered two times. Fixed
- Deleting an entry without entries in the list caused recursive function calling and stack overflow ;-) Fixed!
- Insertion had some bugs which caused the machine to hang or crash. Fixed!

V18.0 - Initial public beta release

1.11 Planings for next releases

ToDo:

- More styles
- Changing the kind of displaying active nodes.
- Selectable key for automatic child selection

ToDo programmers section:

- Implementation of MUIM_NListtree_InsertList, MUIM_NListtree_InsertArray and MUIA_NListtree_SourceArray

and whatever you need...

1.12 Where to get the updates?

The latest release of NListtree is always available at <http://www.aphaso.de>.

1.13 Acknowledgements

Acknowledgements

- Gilles Masson for his great NList.mcc
- Stefan Stuntz for fantastic MUI
- Members of MUI@egroups.com for interesting informations
- Alessandro Zummo for some interesting information about MUI (MUIUndoc) (which does not mean, that NListtree uses undocumented features).
- Henning 'Lemming' Thielmann for interesting informations and suggestions for multiselection and moves/copies
- Pascal Chiozzi for translating the catalog into french
- and all others who have sent bug reports and suggestions

1.14 MagicUserInterface

This application uses

MUI - MagicUserInterface

(c) Copyright 1993/97 by Stefan Stuntz

MUI is a system to generate and maintain graphical user interfaces. With the aid of a preferences program, the user of an application has the ability to customize the outfit according to his personal taste.

MUI is distributed as shareware. To obtain a complete package containing lots of examples and more information about registration please look for a file called "muiXXusr.lha" (XX means the latest version number) on your local bulletin boards or on public domain disks.

If you want to register directly, feel free to send

DM 30.- or US\$ 20.-

to

Stefan Stuntz
Eduard-Spranger-Straße 7
80935 München
GERMANY

1.15 NList.mcc, NListview.mcc, NFloattext.mcc, NListviews.mcp

What Gilles Masson originally wrote to NList:

Copyright

NList.mcc (C) 1996-2001 Gilles Masson

This data is freely distributable, but copyrighted by me. This means you may freely copy them as long as you do not charge more than a nominal fee for copying. This data may be put on PD disks, especially on Fred Fish's AmigaLibDisks and Fred's or Aminet CDROMs. If you want to distribute the data you MUST keep this document with it. All documents, programmes and files must be distributed in their original unmodified form.

The author can not be made responsible for any damage which is caused by using these programs.

This class is FreeWare for end users, but ShareWare for developers : Anyone which make a Shareware or commercial application using it must send me a registered/full working release of the application and updates, there is no other fee (and will not ever).

Anyway, i'll be happy if you send me something... ;-)

The AUTHOR of a program can include the NList.mcc, NListview.mcc, NFloattext.mcc, NListviews.mcp and Install files in its program archive as soon as the above is respected. You can install them from you're own script or use the provided one as you want.

I'm interesting in who is using my classes in his own (free) programs, so don't hesitate to contact me. (just a mail telling what you're doing with NList would be nice :)

Send comments, suggestions or bug reports, or some money ;) to:

Gilles Masson,
Les Balcons d'Antipolis
Bloc C
15, Traverse du Barri
06560 VALBONNE
FRANCE

e-mail: masson@iutsoph.unice.fr

What are NList.mcc/NListview.mcc/NFloattext.mcc/NListviews.mcp ?

NList is a Custom Class for the Magic User Interface (c) by Stefan Stuntz. It's a subclass of area-class which make the same stuffs than the MUI List/Listview class, and more.

NList implement horizontal scrolling, built-in copy to clipboard, classic char precision selection of area, drag and drop and some others.

NListview give NList scrollbars, vertical as Listview, but horizontal too.

NFloattext is like Floattext and do automatic word wrap. The interest to do it with NList is the builtin copy to clipboard.

There is a mui pref class NListviews.mcp with it, which permit the user to choose all default backgrounds, pens and fonts.

1.16 mcc-install

Original copyright notice for MCC-Install from Kai Hofmann:

Copyright

This software and its documentation are copyright 1996-1997,1999 by Dipl.-Inform. Kai Hofmann. All rights reserved!

- Permission for COMMERCIAL USE is only given by an extra available commercial license that must be validated!
Contact me directly for this license, because it will be individually handed out per your needs!
- Permission is hereby granted, without written agreement and without license, to USE this software and its documentation for any NON-COMMERCIAL purpose, provided that the above copyright notice and the following paragraph appear in all copies of this software (Non-commercial includes Giftware!).

Please send me a full version of your product at no cost including free updates!
