AddDataTypes

Roland Mainz and Christian Buchner

	COLLABORATORS		
	TITLE : AddDataTypes		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Roland Mainz and Christian Buchner	February 24, 2025	

	REVISION HISTORY		
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

AddDataTypes

1.1   AddDataTypes/Main

AddDataTypes V45.2

Main DataTypes Preferences

Written (c) 1994/1995 by Christian Buchner

Updated by Roland 'Gizzy' Mainz since 1996/1997

This software is freely distributable.

This project is part of the "DataTypes Replacement Project", which

includes new "AddDataTypes", new "datatypes.library" and a new

"MultiView" (GMultiView).

Introduction What the heck is this good for?

Installation It's so easy.

Usage Just like the old command!

Edit Edit the descriptors

Known problems Useful to know about.

History History

ToDo/Future Future

The authors How to contact the developers.

Support for Tool-Nodes New feature !

DataTypes DataTypes currently available.

1.2   AddDataTypes/Introduction

Introduction

This is a new AddDataTypes command. It offers more flexibility than the previous versions offered by Commodore.

1) It offers the possibility to use subdirectories in DEVS:DataTypes/. Finally, no more chaos in your datatypes drawer.

2) It fixes some memory leaks of the old AddDataTypes command when replacing datatype descriptors.

3) It is 100% compatible to the original versions V39 and V40

4) It includes a preferences editor.

1.3 **AddDataTypes/Installation**

Installation

Open a Shell and type the following:

Rename "C:AddDataTypes" "C:AddDataTypes_cbm"

Copy CLONE "AddDataTypes" "C:AddDataTypes"

That's all.

Then you may add directories in you "DEVS:DataTypes/"-directory, where you can move the matching descriptors.

Don't you think this is much more pleasant than the old chaos in your datatypes drawer?

1.4 **AddDataTypes/Usage**

Usage

The AddDataTypes command will install the descriptor files for your datatypes in your memory. The datatypes.library requires this information in order to recognize the file types. If you forget to call AddDataTypes in your Startup-Sequence, the datatypes.library WILL NOT OPEN!

The AddDataTypes command will NOT remove any datatype descriptor from memory, even if it has been deleted from your DEVS:DataTypes drawer. But it will replace the datatype descriptors in memory, if the descriptor to load has the same name and different settings.

The new AddDataTypes command will exactly behave like the old command, except recursive directory scanning and support for Tool Nodes.

Usage from Workbench:

Just double-click a datatype descriptor icon and the descriptor will be installed/replaced in memory. This requires that the AddDataTypes command has been set als icon's default tool.

Usage from CLI:

The AmigaDOS Template is FILES/M,QUIET/S,REFRESH/S,EDIT/S,USE/S,SAVE/S

As FILES you may specify one or more directories and/or filenames that will be loaded/replaced as datatype descriptors. The directories will be scanned recursively for datatype descriptors, thus allowing you to sort your descriptors into Subdirectories.

The QUIET option will suppress any error messages and other output.

The REFRESH option will automatically rescan DEVS:DataTypes for descriptors, if the date of the DEVS:DataTypes drawer has changed. Also see known problems. At bootup time the directory will ALWAYS be scanned as no previous date is yet known. If you specify REFRESH, the FILES parameter will be ignored.

The EDIT option invokes the editor. This overrides the USE and SAVE options (selected by the GUI).

The USE option means: Use the descriptors (default).

The SAVE option means: Save all active descriptors to "DEVS:DataTypes/".

Examples:

If you want to force a rescan, just type

AddDataTypes DEVS:DataTypes

In your startup-sequence you may want to use

AddDataTypes REFRESH

To install one or more specific datatypes, type e.g

AddDataTypes DEVS:DataTypes/Graphics/GIF SYS:Storage/DataTypes/Sound/WAVE

To invoke the editor, type e.g (see Edit section)

AddDataTypes EDIT

To save all currently active descriptors to "DEVS:DataTypes/", type

AddDataTypes SAVE

A perfect do-nothing command is

AddDataTypes

without parameters. Either specify REFRESH,some FILES, EDIT or SAVE okay?

1.5   AddDataTypes/Known problems

Bugs and Known problems

*****

***** *************)

********* * ***********************) *

*** ******* ******************************) *

*** (***) *

(***) *

(*** ***

(*** (O) ***

(***)

(***)

(***)

(******* ************************************)

(*** *** ** *** * *******

(*** *** *** *** * *******

(*** *** * * * * *

(* **** * * * * ***

(* *** *** * * * ***

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

- Not really a bug, but a misfeature of AmigaDOS:

The REFRESH option will check if the date of the directory DEVS:DataTypes has been changed and will rescan the datatype descriptors in this case. The date will be changed if you modify or replace a datatype descriptor in DEVS:DataTypes.

BUT: If you modify the contents of a subdirectory of DEVS:DataTypes, the date of DEVS:DataTypes itself will _NOT_ change. So the REFRESH option will not work. Use "AddDataTypes DEVS:DataTypes" in this case to force a rescan.

Will be fixed.

- The error-handling isn't perfect:

The current AddDataTypes-Version does not report all errors which may occur (e.g corrupt descriptor files etc.).

The following list describes some errors which are currently not notified to the user:

- Low memory

Will be fixed.

- Low memory causes problems,

descriptors won't be saved, or the editor may not show all descriptors etc.

Will be fixed.

- The file version (e.g. the generic FVER chunk is ignored) will not be written when descriptors are

saved.

Will be fixed.

- If there are no changes, and "Save" was selected, the complete descriptor list is saved.

May or may not be fixed; it would be difficult to keep track of the changes in a descriptor, and

if it resides already in DEVS:DataTypes/ in this version or not.

1.6 AddDataTypes/The Authors

The Authors

The future development of "AddDataTypes", "datatypes.library" and "GMultiView" is done by

Roland Mainz

Hohenstaufenstraße 8

52388 Nörvenich

Germany

Tel.: +49-2426-901568

EMAIL: GISBURN@W-Specht.rhein-ruhr.de

or

EMAIL: Reinhold.A.Mainz@KBV.de

If you want to contact Christian Buchner:

Christian Buchner

Steinbergstraße 2

82110 Germering

Germany

Tel.: +49-89-849868

EMAIL: flowerp@eikon.e-technik.tu-muenchen.de

<flowerp> on irc #amiga and #amigager

1.7 AddDataTypes/DataTypes

This is a list of all the datatypes and utility programs I was able to get hold of. The datatypes have been renamed to BASE-NAME_dtVersion.Revision and archived with the LZX archiver. Most files have been taken from Aminet. The BBS file comments are German and contain the names of the authors.

<Not written yet, sorry>

You may find a good list in http://www.execpc.com/%7Einnuendo/amiga/os/datatypes.

1.8 AddDataTypes/Tool Nodes

New: Tool Nodes

Tool nodes are a currently unused datatypes feature. Each datatype can have a set of tools which can use the given datatype.

These tools are seperated in the categories Info, Browse, Edit, Print, Mail and Misc.

Example:

The IFF ILBM DataType would have the following tools (in it's datatypes descriptor):

Info:

C:List -- Shell command list, informs the user about file size etc.

SYS:Utilities/GMultiView -- Extensive information

Browse:

SYS:Utilities/MultiView -- Shows the picture

SYS:Utilities/GMultiView -- Shows the picture

SYS:Utilities/Display -- Shows the picture

Edit:

DPaintIV:DPaint -- DeluxePaint

Print:

SYS:Utilities/MultiView -- Is able to print the picture

SYS:Utilities/GMultiView -- Is able to print the picture

DPaintIV:DPaint -- DeluxePaint

TurboPrint:TP -- Turbo Print

Mail:

The mailto:-device etc.

SendMail

Misc:

--

How to edit/add these descriptors ?

See Edit section

Which tools supports this:

Currently, only GMultiView supports this.

GMultiView is available (demo version) in the aminet in the util/dtype-directory.

1.9 AddDataTypes/Edit

AddDataTypes allows you to edit the descriptors

Currently, you can only edit the tool nodes, later versions were able to do more,

see ToDo section.

Type in a Shell "AddDataTypes EDIT", and you'll get the editor.

The "DataType"-Listview gadget shows you all available datatypes in the system.

The "Tools"-Listview shows the tools defined for the selected datatype, if no datatype is selected, the listview is disabled (ghosted).

The "Which"-Cycle-Gadget, the "Program"-String-Gadget and the "Launch as"-Cycle-Gadget describes the tool, it's category and it's launch method.

The tools are seperated in six categories (selected by the "Which"-Gadget),

Misc - A category which does not match the others below

Info - This tool shows information like statistics etc. about the data

Browse - This tool is able to show the data in it's specific form

Edit - This tool is able to edit the data

Print - This tool is able to print the data

Mail - This tool is able to mail the data

There are currently three launch methods defined for the tools:

Shell - Start the tool from shell

Workbench - Start the tool from Workbench

ARexx - Start the tool using ARexx

On startup, these tools may get the data as an argument.

The menu offers the "standard" preference items, as

Project/Open... - Open one or more descriptors, or scan the whole directory for them

/Save as... - Save all descriptors to the given directory.

/Quit... - Quit (Cancel)

Project/Reset to default - Reset the descriptor list, only the ascii, iff,

binary and directory descriptors will be available

/Last Saved - Reset the descriptor list and load the descriptors from DEVS:DataTypes

/Restore - Restore list (use currently active descriptors again)

Settings/Create Icons ? - Create icons when saving

1.10   AddDataTypes/ToDo

Future versions will implement:

- Fixing the known bugs.

- Add tooltypes support

- More things to edit in the editor,

- ASL File requester to select the tool's path.

- the descriptor data itself,

- the so-called DTTG (default tags, preferences for datatype classes)

and

- DataType-specific preferences. This editor will be the

preferences editor for both datatypes.library and the

local classes (e.g. #?.datatype)

- More robust code.

- Online-help.

1.11 AddDataTypes/History

History:

V41.4

- First Aminet release

V41.5

- Fixed silly bug in WB startup code, which causes an endless loop.

(But this WB startup code was never called, because cres.o was linked

to the executable.)

Thanks to Gunther Nikl (gnikl@informatik.uni-rostock.de)

Fixed.

- Increased the size of tool path length from 256 up to 4096.

(mail me if you need more :-))

- Increased some internal write buffers

- Recompiled with datatypes.library V45 includes

- The ASL file requester is now opened relative to the parent window.

V41.6

- Fixed a bug in the saving routines: The comparisation mask was not written completely,

this caused problems with some descriptors.

Thanks to Joern Krueger (jkrueger@muenster.de) for reporting this bug.

Fixed.

V45.1

- Set the projects version to 45.

This is the first release for the datatypes.library V45 distribution.

- The stack size of icons written was 0, fixed to 8192 bytes.

Thanks to Joern Krueger (jkrueger@muenster.de) for reporting this bug.

Fixed.

- The GUI has been rewritten;

- The window's GUI (gadgets, menu etc.) is now correctly blocked when executing

command like save, load etc.

- The window now has a zoom gadget.

- GUI is now prepared for online-help (menu help; gadget help)

- The "Tools/Program" String-Gadget now has a file selection icon (which opens

a given tool).

Thanks to Joern Krueger (jkrueger@muenster.de) for this idea.

Selecting multiple tools copies the current tool's attributes (which action, launch mode)

into new tool entries created for the selected tools.

- Now includes a Add/Delete buttons for descriptors. Note that the descriptors

were _NOT_ deleted from disk when using the "Delete"-button, they'll only be

removed from the current list of descriptors.

- Fixed an enforcer hit (in GadTools) when the descriptor list was changed completely,

but the gadgets holds references to the old list.

Fixed.

- Fixed the bug that existing icons were overwritten with the "default" DataTypes-Descriptor

icon. Now AddDataTypes creates icons when they don't exists, but does not touch the

existing ones.

Thanks to Joern Krueger (jkrueger@muenster.de) for reporting this bug.

Fixed.

- If a shared library cannot be opened or has a version which is too old, AddDataTypes

will now notify the user.

- If a descriptor couldn't be saved, AddDataTypes now notifies the user.

- Added ASL filter code which displays only IFF DTYP files now (temporary disabled) when

opening descriptors.

- Added AppWindow. If AddDataTypes does not reside on the workbench screen, an AppIcon

is used instead.

- If the icon is dropped on the descriptor list will select the matching descriptor.

If the icon is a IFF DTYP one, AddDataTypes will ask to add this descriptor or to

jump to the matching descriptor.

- If the icon is dropped on the tools list, a new tool entry is created (icon name

as program).

- If another location is used, or the icon was submitted through an appicon,

AddDataTypes asks the user what to do with the icon.

LAST MINUTE NOTE: feature temporary disabled; sorry.

V45.2

- Fixed the bug that AddDataTypes requests locale.library with a minimum version

of V39. Now this is correctly set to V38, and AddDataTypes won't fire up the

"Lib not found"-requester if locale.library wasn't found (not a MUST-BE-THERE

resource).

Fixed.

- Added basic descriptors for all GID_#? types (e.g. system, text, document, sound,

instrument, music, picture, animation and movie).

- Some internals prepared for "objecttools.library".

- Added WB tooltype support.

- Added "Prefs"-like WB icon.

- Added check for illegal characters in the descriptor to get rid of the saving problem

(e.g. when a descriptor's name contains ':', '/' etc.).

Seems that AddDataTypes needs an internal code cleanup...