

**Deft II v1.6**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> Deft II v1.6		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		November 24, 2024	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Deft II v1.6</b>	<b>1</b>
1.1	Deft II v1.6 . . . . .	1
1.2	Introduction . . . . .	1
1.3	Installation . . . . .	2
1.4	Using . . . . .	2
1.5	Tutorial . . . . .	3
1.6	How I use myself Deft II... . . . .	4
1.7	The GUI . . . . .	4
1.8	The 'Default tools' group . . . . .	4
1.9	The list . . . . .	5
1.10	The string 'Old' . . . . .	5
1.11	The popobject gadget 'New' . . . . .	5
1.12	The 'Delete' button . . . . .	5
1.13	The 'Paths' group . . . . .	5
1.14	The list . . . . .	6
1.15	The popasl gadget . . . . .	6
1.16	The 'Delete' button . . . . .	6
1.17	The text gadget 'File' . . . . .	6
1.18	The 'Deft II controls' group . . . . .	6
1.19	The 'GO !!!' button . . . . .	6
1.20	The 'Stop' button . . . . .	7
1.21	The 'Save prefs' button . . . . .	7
1.22	The 'About' button . . . . .	7
1.23	The 'Quit' button . . . . .	7
1.24	The ARexx port . . . . .	7
1.25	Basic MUI ARexx command . . . . .	8
1.26	ARexx command 'number_paths' . . . . .	8
1.27	ARexx command 'number_default_tools' . . . . .	8
1.28	ARexx command 'add_path' . . . . .	8
1.29	ARexx command 'add_default_tool' . . . . .	8

---

1.30	ARexx command 'delete_path' . . . . .	9
1.31	ARexx command 'delete_default_tool' . . . . .	9
1.32	ARexx command 'get_path' . . . . .	9
1.33	ARexx command 'get_old_default_tool' . . . . .	9
1.34	ARexx command 'get_new_default_tool' . . . . .	9
1.35	ARexx command 'save_prefs' . . . . .	9
1.36	ARexx command 'go' . . . . .	10
1.37	ARexx command 'loose_modifications' . . . . .	10
1.38	The sources . . . . .	10
1.39	Bugs . . . . .	10
1.40	History . . . . .	11
1.41	To do . . . . .	11
1.42	Distribution . . . . .	11
1.43	The author . . . . .	13
1.44	The acknowledgements . . . . .	13
1.45	Index . . . . .	14

---

# Chapter 1

## Deft II v1.6

### 1.1 Deft II v1.6

```
*****
```

```
                Deft II (v1.6)
Utility to replace icon default tools
Archive of November 1994, the 15th
© Copyright 1994, Lionel Vintenat
```

```
*****
```

```
Introduction
Installation
Using
The GUI
The ARexx port
The sources
Bugs
History
To do
Distribution
The author
The acknowledgements
```

### 1.2 Introduction

Let's have a little trip in the past. Some times ago, I found on a fish disk an excellent little program named Deft (© 1992, Gary Smith). Its goal was very simple : change the icon default tools. Let's take a using example. Suppose that you get used to reading your documentations with More program. In this case, all your documentation icons have as default tool something like "SYS:Utilities/more". And then, a nice day, you discover the fabulous Most. And of course, starting this moment, you don't want to hear about More ever (and I agree ! :)). But change all your default tool fields of your documentation icons may be a HUDGE task if your hard drive is a big one. It's at this point that Deft comes up. You give it a path, an old

---

default tool, a new default tool, and goooo, it makes all the replacements on your hard drive. Briefly, Deft is (for my point of view) a typical PD program : too small to be commercial, but genius for the idea, and dramatically useful (we don't need it often, but when we need it, we are REALLY happy to have it !!!).

You probably guess it, Deft II is a new version (starting from nothing as I hadn't Deft sources) of Deft. This program seemed to be undevelopped for a long a time, so I allow myself to rewrite it, adding -I hope- a lot of enhancements, namely :

- a MUI GUI with all the consequent convenience for the user (keyboard shortcuts, appwindows, etc)
- multi-paths handling for scanning
- multi-default tools for replacing
- wildcard using for the default tools to replace
- preferences saving
- ARexx support
- online help

About this documentation reading, read completely the chapter using . All that you need to know is there. The chapter about the GUI is in fact used for the online help, and thus, may be used as a reference document. Finally, if you are an ARexx wizard, have a look on the chapter ARexx .

WARNIGN : Deft II needs 2.0 Workbench or newer to run, and MUI v2.2 properly installed.

## 1.3 Installation

If you have the Installer program, just let you guide !

If it isn't the case, don't panic ! Deft II is a very simple program which can be installed easily and without pain manually :

- 1) Copy all the files "Deft II", "Deft II.info", "Deft II.prefs", "Deft II.guide" (the one which corresponds to your language), and "Deft II.guide.info" inside a SAME directory of your choice.
- 2) Copy "DeftII\_Test.rexx" inside REXX:.
- 3) Copy the "DeftII.catalog" file which corresponds to your language if it is available and if your Workbench supports localization (normally only Workbench versions newer or equal to 2.1 support it) inside the system directory "LOCALE:Catalogs/votre\_langue/".
- 4) (optional) Copy the "Sources/" directory where you want.

## 1.4 Using

---

Let's see first a little tutorial about Deft II using.  
Then let's follow with this fundamental and serious question :  
But how does the author himself use his program ? :)

## 1.5 Tutorial

Open the unique window of Deft II. This one can be split into three parts :

- on the left the 'Default tools' group
- on the right the 'Paths' group
- on the bottom the text gadget 'File' and the 'Deft II controls' group

The 'Default tools' group is used to define which default tools Deft II will have to replace, and by which new default tools it will have to replace them.

Each old/new default tool pair is displayed in this order inside the list. To enter such a pair, use the two string gadgets 'Old' and 'New'. Note that the pair is enregistered when (and only when) you strike the enter key inside the text gadget 'New'. Note also that Deft II doesn't accept a pair if the old default tool is empty or if it already exists a pair with the same old default tool. But it's possible to enter a pair with the new default tool empty. This pair will be then displayed in italic inside the list to signal you that the definition of this pair is incomplete.

The old default tool may have wildcars (not the new one of course !). For instance, "#?more#?" is a possible old default tool. During the replacement stage, Deft II will search among all the project icons located inside the paths to scan if one has a default tool which matches this pattern, and will perform, in case of matching, the replacement. You can thus replace very easily the "c/more", "c:more" and other ":c/MuchMore" by "c:Most" ! Better, you can without any trouble ask Deft II to replace all the "#?guide#?" by "c:AmigaGuide" because Deft II check before a replacement that it isn't going to replace a string by itself. So, a default tool like "C:amigaguide" (note that Deft II is case insensitive), won't be replaced whereas ":c/AmigaGuide" will.

The popup gadget on the right of the string gadget 'New' opens a little list where all the new default tools already entered in Deft II are : you can thus select one without needing to type it again. But the first time, you can use the popfile gadget on the right of the popup gadget. This opens a file requester where you can choose a new default tool.

Double-clicking on a pair in the list enables you to edit it.

You can drop on the list and on the 'Old' string of the 'Default tools' group some project icons. If you do, the default tools of these icons will be immediately added (if they weren't already there of course) in the list. In these case, they will appear in italic because they haven't yet a corresponding default tool.

Finally, you can drop on the 'New' string of the 'Default tools' group a tool icon. If you do, this one will appear in this string. This does the same as the "popfile" gadget on the right of this string.

The 'Paths' group is used to define the paths to recursively scan during the replacement stage, paths which are displayed in a list. If you drop an icon to this group, the path of the file or of the drawer is added

---

to the paths list. Double-clicking on a path in the list enables you to edit it.

Finally, you can use the 'Deft II controls' group to run the default tools replacement stage (button 'GO !!!'), to stop the replacement stage at any moment (button 'Stop') or to save the preferences (button 'Save prefs'), preferences which hold the two list contents. During the replacement stage, Deft II displays inside the text gadget 'File' all the files whose icon is modified by Deft II.

## 1.6 How I use myself Deft II...

I personally configured Deft II to use it in the case of a replacement on my hard drive. In other words, the preferences I saved hold all my hard drive partitions inside the paths list.

However a frequent case when the default tool replacement is needed appears during a new program test (thing I usually do in RAM: or SD0:). Indeed, it's often the case that, when I click on the installation icon or on the documentation icon, the Workbench doesn't find the right default tool. To solve this problem, I run before starting my tests Deft II by a hotkey from ToolManager. Deft II starts then iconified (thing to configure with MUI-Prefs if you are a register user). Then for each new test in RAM: or in SD0: (i.e. for each new archive unarchived), I run, always with an hotkey from ToolManager (I love this program :)), a little ARexx script, "DeftII\_test.rexx". This one clears the path list of Deft II replacing them by "RAM: and "SD0:", and just after runs the replacement stage (so located in RAM: and SD0:). Once this one finished, it re-builds the path list as it found it at starting. In other words, thanks to this little ARexx script and to ToolManager, I can perform a very fast replacement (RAM operation and Deft II is always iconified) of the default tools for each new archive during its test stage, and all this without loosing my configuration of Deft II, this last one sticked to my hard drive partitions.

So I advise you to try this ARexx script, even if you aren't a wizard of this language (I put all the required comments in order you can adapt it to your needs). For my point of view, it completely automates Deft II using.

To help you, I provided in this archive my own Deft II preferences file.

## 1.7 The GUI

- The 'Default tools' group
- The 'Paths' group
- The text gadget 'File'
- The 'Deft II controls' group

## 1.8 The 'Default tools' group

---



Manages the old/new default tool pairs.

If you drop a project icon on the list or on the 'Old' string of this group, its default tool will be automagically added to this list (if it isn't already there of course) as "old default tool". It hasn't yet a corresponding "new default tool".

If you drop a tool icon on the 'New' string of this group, this one will appear in this string.

```
The list
The string 'Old'
The popobject gadget 'New'
The 'Delete' button
```

## 1.9 The list

Displays all the old/new default tool pairs known by Deft II. If a pair hasn't a new default tool, it is displayed in italic.

Double-click on a pair in this list to edit it.

### 1.10 The string 'Old'

Enter in this string an old default tool to replace.

### 1.11 The popobject gadget 'New'

Enter in this string by which new default tool Deft II must replace the associated old default tool. Strike the ENTER key inside this string to validate the current edited pair.

If you click on the popup image on the right of the string, it opens a list of the new default tools already entered in Deft II. Double-click on one entry of this list to select it.

If you click on the popfile image on the right of the popup image, this opens a file requester to choose a new default tool to put in the string.

### 1.12 The 'Delete' button

Deletes the active pair from the list.

### 1.13 The 'Paths' group

---

Manages all the paths to scan.

If you drop an icon inside this group, its path will be automagically added to this list (if it isn't already there of course).

- The list
- The popasl gadget
- The 'Delete' button

## 1.14 The list

Displays all the paths known by Deft II.  
Double-click on a path in this list to edit it.

## 1.15 The popasl gadget

Enter here a path to add to the list. You can either type manually this new path, or choose it with a file requester by clicking on the popasl gadget on the right of this string.

## 1.16 The 'Delete' button

Deletes the active path from the list.

## 1.17 The text gadget 'File'

During the replacement stage, displays all the files whose icon is modified by Deft II.

## 1.18 The 'Deft II controls' group

- The 'GO !!!' button
- The 'Stop' button
- The 'Save prefs' button
- The 'About' button
- The 'Quit' button

## 1.19 The 'GO !!!' button

---

Starts the replacement stage : Deft II recursively scan all the given paths, and change the "default tool" field of each icon where this field mathes an "old default tool" to the "new default tool" associated.

The old default tool may be an AmigaDos pattern and Deft II checks before the replacement that the current "default tool" field isn't the same than the "new default tool" associated.

## 1.20 The 'Stop' button

Enables you to stop a any moment the replacement stage.  
Without effect in the other cases.

## 1.21 The 'Save prefs' button

Saves Deft II preferences in the current path under the name "Deft II.prefs".

## 1.22 The 'About' button

Prompts a requester which talks you about me :).

## 1.23 The 'Quit' button

Quits Deft II.  
If a file requester is still opened, Deft II will complain.  
If the preferences have been modified without being saved, Deft II will ask you confirmation.

## 1.24 The ARexx port

The Arexx port of Deft II is named DEFT\_II.n where n is 1, 2, etc according to the number of Deft II you have run. The commands of this port are the following :

```
quit          FORCE/S
activate
deactivate
show
hide
info          ITEM/A
help          FILE/A
```

```
number_paths
```

---

```

number_default_tools
add_path           PATH/A
add_default_tool   OLD/A,NEW/A
delete_path        PATH/N/A
delete_default_tool DEFAULT_TOOL=DT/N/A
get_path           PATH/N/A
get_old_default_tool OLD_DEFAULT_TOOL=ODT/N/A
get_new_default_tool NEW_DEFAULT_TOOL=NDT/N/A
save_prefs
go
loose_modifications

```

Note that none of the Deft II own ARexx commands prompt the apparition of requesters inside Deft II; this sometimes differs from the same action this time called from the GUI (example : "'go' ARexx command and 'GO !!!' button).

## 1.25 Basic MUI ARexx command

This ARexx command is always present in all the MUI applications. So refer you to the MUI (user) documentation for further details.

## 1.26 ARexx command 'number\_paths'

Returns the number of paths present in the list of the 'Paths' group.

## 1.27 ARexx command 'number\_default\_tools'

Returns the number of old/new default tool pairs present in the list of the 'Default tools' group.

## 1.28 ARexx command 'add\_path'

Adds the given path in the list of the 'Paths' group.

A return code of 10 indicates that the given path was already there.

## 1.29 ARexx command 'add\_default\_tool'

Adds the given old/new default tool pair to the list of the 'Default tools' group.

A return code of 10 indicates that the given pair was already there.

### 1.30 ARexx command 'delete\_path'

Deletes from the list of the 'Paths' group the path whose number is given as parameter (the numbering starts from 0).

A return code of 10 indicates that the given number doesn't correspond to any path (number too large or negative).

### 1.31 ARexx command 'delete\_default\_tool'

Deletes from the list of the 'Default tools' group the old/new default tool pair whose number is given as parameter (the numbering starts from 0).

A return code of 10 indicates that the given number doesn't correspond to any pair (number too large or negative).

### 1.32 ARexx command 'get\_path'

Returns the path whose number is given as parameter (the numbering starts from 0).

A return code of 10 indicates that the given number doesn't correspond to any path (number too large or negative).

### 1.33 ARexx command 'get\_old\_default\_tool'

Returns the old default tool whose number is given as parameter (the numbering starts from 0).

A return code of 10 indicates that the given number doesn't correspond to any old default tool (number too large or negative).

### 1.34 ARexx command 'get\_new\_default\_tool'

Returns the new default tool whose number is given as parameter (the numbering starts from 0).

A return code of 10 indicates that the given number doesn't correspond to any new default tool (number too large or negative).

### 1.35 ARexx command 'save\_prefs'

Makes Deft II save your preferences.

A return code of 50 indicates that an error occurred during the preference saving, and that this one doesn't happen correctly.

---

### 1.36 ARexx command 'go'

Runs the replacement stage of the default tools according to the current preferences.

A return code of 10 indicates that some incorrect paths were paths during this stage, and that these ones were removed from the list of the 'Paths' group. 5 is added to this return code if this stage was stopped by the user before its normal end. Finally, 100 is added to this return code if a serious error has prevented Deft II from the access of a file or a directory. In this case, Deft II immediately stops its activities while freeing all the ressources it owns.

To sum up, the possible return codes are 5, 10, 15, 100 and 110.

### 1.37 ARexx command 'loose\_modifications'

Makes Deft II forget that some modifications were made to the preferences. This allows doing such modifications from a script but without the user being annoyed by a confirmation requester when he will want to quit Deft II.

### 1.38 The sources

All the sources of Deft II are provided. So you can recompile this program without any trouble. For this, you need Mac2E v4.0+ and autobuild v1.0+ in addition of Amiga E v3.0+. If you followed the installation instructions of these programs, the command "autobuild" would be enough.

However don't forget that modifications to the files of this distribution is only allowed for your strictly personal usage (see distribution) ! If you find a bug and fix it, please, write to me in order I make a new version, thanks.

The main reason for which I provided all the sources is of course that they can be used as examples for other programmers. I myself took a lot of benefit from other programmer sources, so I hope that the chain will continue :).

Finally, I precise that I joined the ".cd" and ".ct" files to let you re-build the catalog for your language. If you make such a catalog, don't forget to send it to me in order I can include it in the future released versions, thanks.

### 1.39 Bugs

None known.

---

## 1.40 History

- Version 1.0 : - First released version
- Version 1.1 : - re-programmed for Amiga E v3.0a
  - localization problem resolved (before some strings didn't appear when default English language was used)
- Version 1.2 : - optimisation of the replacement stage with the ExAll() function => 4 times faster !
  - in case of memory starvation during this stage, Deft II now frees all the allocated ressources
- Version 1.3 : - addition of the popfile gadget to choose a new default tool
  - it's possible to drop on the 'New' string a tool icon to put this tool in this string
  - 2nd released version
- Version 1.4 : - german catalog added
  - recompiled with Amiga E v3.0b
  - better localization handling (internal)
  - more general preferences file
- Version 1.5 : - enforcer hits removed : nothing serious (only reads) but<<nothing should be causing enforcer hit>>(enforcer readme)
- Version 1.6 : - enforcer hits removed (grrrr I hope they were the last ones this time...) : still nothing serious but...
  - bug fixed : at startup time, Deft II could hang when an MUI error occurred
  - recompiled with Amiga E v3.0b

## 1.41 To do

I have in mind to write a commodity in addition of Deft II which would patch the system to change in real time the default tools. Of course, this one would use the preferences de Deft II. But for the moment, it's just an idea, not a project.

Apart this I don't see anything to add to Deft II, except fixing bugs of course. But I'm opened to every reasonable suggestion (see the author ).

## 1.42 Distribution

All the files of this distribution stay under the copyright of the author (Lionel Vintenat). You are allowed to modify them only for your STRICTLY PERSONAL usage.

The only exceptions are the "Readme.mui" file and the icons.

This archive can be freely distributed by any thinkable ways (ftp server, BBS, public domain collection, etc), as long as the two following conditions are respected :

- 1) No person gains ANY benefit from this distribution. In
-

particular, if Deft II is spread on a floppy disk, this one can't be sold for more than 4\$ US (or equivalent), and if it is spread on a CDROM, this one can't be sold for more than 30\$ US (or equivalent). No other type of sale (with benefit) can be made IN ANY CASES without the author's authorization. The only exceptions are the Fred Fish's and aminet CDROMs, who they (and only them !) can include Deft II in their collections without asking me the permission first. In particular, this excludes DEFINITELY to France Festival Distribution the right to distribute Deft II (I insist HEAVILY on this point...). But maybe this won't annoy too much Mr Serge Hammouche, who doesn't hesitate to call (openly on some French nets) French PD programmers incapable...

2) This archive is distributed IN IT'S ENTIRETY, and without MODIFICATIONS compared to the original version on aminet. This means in particular that if you make a translation of the documentation or of the catalog in a new language, or if you fix some bugs and re-compile the executable, you MUST send them to me in order I EVENTUALLY (pretty sure in fact for translations) redistribute a new version of this program. The file structure this archive must have is given in the "ReadMe.first" file of this distribution.

Any distribution of Deft II which doesn't respect the two previous conditions without my authorization is ILLEGAL.

All the beginning of this paragraph may seem very strict, even close to the paranoia, but knowing the dubious practices of people like Serge Hammouche who sells for HUDGE prices (under translation excuse) some freely distribuable softwares without even informing the authors, I think it's necessary to protect my rights. I make programs for my pleasure without any claim to earn money, and I'm happy if they may help other persons, but that some people make money from my work : NO ! The previous limitations doesn't aim in any way serious people like Fred Fish, the Montréal Amiga Club, the aminet system operators, or all the clubs of fascinated people, who, them, really support the Amiga domain public. They only aim the organizations with discutable practices like France Festival Distribution.

Moreover, I cannot be held responsible for the use of this program and any damages that it may cause directly or not. I want this to be clear : YOU USE IT AT YOUR OWN RISK !

However, I think I tested and I made people test it enough to say that it doesn't contain any serious bug.

Finally, this program is distributed under giftware concept. In other words, you must send me a gift if you use Deft II ! :)

Indeed, I make programs for my pleasure and by need. My goal is certainly not to earn money with. However to distribute a program ask for some extra work (documentations, installer script, etc), so I'd like to receive a feedback from those who will use Deft II. In fact, any sign of life will be VERY appreciated, even a simple e-mail or a postcard. What I am the more interested in is contact with other persons. But to help you to choose my gift, here are some suggestions :) :

- a (free !) registration to a shareware program
  - one of your realizations (program, module, animation, picture,
-



etc) if it isn't easy to retrieve it from aminet

- some sources
- some E, C or assembly sources which are closed to the programation of the system, or even False or BrainFuck sources (they can be closed of anything, provided they work ! :))
- some money, hummmm, why not ? :)
- your old 1.3 RKM (or better 2.0)

I insist on the fact that it's very frustrating to make the effort to put his program in the freely distribuable domain without never receiving any feedback, just guessing that some people use your program ! So please support the giftware concept, everybody break even : the author is happy to receive a feedback, and it costs almost nothing to the users.

## 1.43 The author

You can reach me by snail-mail at my family address :

Lionel Vintenat  
3 impasse Boileau  
Lotissement Les Termes  
87270 COUZEIX  
FRANCE

You can also reach me on the internet. My e-mail address is vintenat@reseau.onecert.fr. This address will remain reliable until August 1995 included. I prefer by far that you write to me by e-mail than by snail mail.

## 1.44 The acknowledgements

To make this program, I used the following softwares :

- Amiga E v3.0e by Wouter van Oortmerssen
- MUI v2.2 by Stefan Stuntz
- MUIBuilder v2.0B by Eric Totel
- Mac2E v4.0 by ... euh ... myself !
- AutoBuild v1.0 by Jan Hendrik Schulz
- FlexCat v1.4 by Jochen Wiedmann
- Iconian v1.98B by Chad Randall

All helped me, and not to say that were absolutely required, so I thanks their authors.

A very special BIG thank you to :

- Gary Smith (I never meet him, but who knows with the magic internet) for making Deft : he is the person to congratulate for the genius idea of this program

- l'Amiga to be the best home computer

- Wouter van Oortmerssen for his work in the field of compilers (try his FALSE, guaranteed surprise!) in general and for Amiga E in particular

- all those on the French Amiga Mailing list who have helped me
  - all those who write public domain programs in general
-

- Boris Jakubaschk for his comments and for the German catalog
- Xavier Billault for his beta-testing (online !) et his suggestions
- Patrick Tremisi for his beta-testing and his suggestions
- Vincent Thomas for his beta-testing and his suggestions

Finally, thank you to all those who will signal me bugs or suggestions, or who will send me corrections or translations of this documentation (see the author ).

Good PD and ...

NEVER FORGET, ONLY AMIGA MAKES IT POSSIBLE !

## 1.45 Index

- ARexx command 'add\_default\_tool'
- ARexx command 'add\_path'
- ARexx command 'delete\_default\_tool'
- ARexx command 'delete\_path'
- ARexx command 'get\_new\_default\_tool'
- ARexx command 'get\_old\_default\_tool'
- ARexx command 'get\_path'
- ARexx command 'go'
- ARexx command 'loose\_modifications'
- ARexx command 'number\_default\_tools'
- ARexx command 'number\_paths'
- ARexx command 'save\_prefs'
- Basic MUI ARexx command
- Bugs
- Deft II v1.3
- Distribution
- History
- How I use myself Deft II...
- Installation
- Introduction
- The 'About' button
- The 'Default tools' group
- The 'Deft II controls' group
- The 'Delete' button
- The 'Delete' button
- The 'GO !!!' button
- The 'Paths' group
- The 'Quit' button
- The 'Save prefs' button
- The 'Stop' button
- The acknowledgements
- The ARexx port
- The author
- The GUI
- The list
- The list
- The popasl gadget
- The popobject gadget 'New'

---

---

The sources  
The string 'Old'  
The text gadget 'File'  
To do  
Tutorial  
Using

---