

RAPrefsMUI

Jaca/Dreamolers-CAPS

COLLABORATORS

	<i>TITLE :</i> RAPrefsMUI		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Jaca/Dreamolers- CAPS	July 10, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	RAPrefsMUI	1
1.1	ReqAttackPrefsMUI guide by Jaca/D-CAPS'99	1
1.2	Introduction	1
1.3	Installation	2
1.4	Features	2
1.5	Legal stuff	2
1.6	Author	3
1.7	Requirements	3
1.8	How does it work?	4
1.9	Main window	4
1.10	Entry's age example	5
1.11	Edit window	5
1.12	Important release notes	7
1.13	Bugs, limitations, future versions	8
1.14	Thanks	8
1.15	History	8
1.16	ReqOFF patch	9

Chapter 1

RAPrefsMUI

1.1 ReqAttackPrefsMUI guide by Jaca/D-CAPS'99

ReqAttackPrefsMUI by Jaca/Dreamolers-Clever Amigants Polish ↔
Society
©1999-2000 by Dreamolers-CAPS

Introduction

Requirements

Features

Installation

ReqOFF

Prefs idea

Thanks

Main window

Author

Edit window

Legal stuff

About this release

Bugs/To do

History

1.2 Introduction

Somewhere in Sep 1999 Cyran/D-CAPS copied his new downloads from Aminet to my harddisk... I found reqattack.lha and after some decrunching I started ReqAttack. It was really amazing (nice difference between this and old ARQ), but the configuration method - script language - simply took too much time to get what I wanted. Well, after a few more minutes I decided to write a prefs program, which would be more friendly to the user. RAPrefsMUI is a script file generator for ReqAttack 1.2 or greater. This program is a result of hours of hard work and I hope you'll appreciate it by registering. The 1st version of this program, ReqAttackPrefs, which can be found on Aminet (util/wb) had an awful GUI (IMHO) and its internal structures made future developing almost impossible. I've decided to write a completely new program. Please read [features](#) part to see what I have managed to change...

1.3 Installation

- * replace your old ReqAttack prefs file with the one from this [↔](#) release
- * copy RAPrefsMUI wherever you like (I advice sys:prefs)
- * copy RASoundDaemon to WBStartup (needed only when you want to use sound with your requesters) (you can also put it to c: and run it from startup-sequence)
- * copy ReqOFF to C: and read
ReqOFF
chapter

1.4 Features

- * NEW: Possibility to switch off some requesters with ReqOFF!
- * Covers most of ReqAttack's check items
- * Covers all ReqAttack's 1.2 configuration items, including sound!
- * Communicates with ReqAttack via ReqAttack Warp Core port
- * Dynamic memory allocation using memory pools
- * Nice and easy to use GUI written in Magic User Interface
- * Reads and saves reqattack.prefs files (no more *rap files!)
- * Bubble help with most gadgets
- * Own RGB Display Class in color sections, which remaps colors to nearest color in palette if there's no free pens
- * Faster, better and easier to use than old version!
- * reqattack.prefs files are now much smaller (in my case it's 12kb to 3kb!)

1.5 Legal stuff

Well, I've decided to make this program CARD/MAILWARE. If you like this program (or not), send me a mail with:

-your name and email address

-some info about the Amiga you use

-comments and bug reports (I'm sure there's some:(

This shouldn't be a big problem and by doing that you'll give me some moral support to work on future versions. For adress see

Author
part.

Important:

- * I take no responsibility for any data loss caused by ReqAttackPrefs/ReqAttackPrefsMUI.
- * this program is NOT a Public Domain, All Rights Reserved.
- * if you want to include this program in some commercial releases (like CoverCD's), it's OK with me, but you need to email me 1st!!!
- * you may spread this archive, but leave it untouched!

1.6 Author

My addy is:

Jacek Piszczek
ul. Dobra 23
87-100 Torun
POLAND

My email is:

jacadcaps@poczta.onet.pl
or jaca@loiv.torun.pl (please use the 1st one)

1.7 Requirements

Software required to run RAPrefsMUI:

- * AmigaOS 3.0+
 - * MUI 3.6+
 - * ReqAttack 1.2+ (needs be running while using RAPrefs!)
 - * PNG datatype to display title logo
-

1.8 How does it work?

Few things have changed from ReqAttackPrefs 1.17 to RAPrefsMUI... Now we've got a little bit different structure types: * DEFAULT or SCREEN* SUBREQUESTER or SUBPROGRAM Let me explain this to you: DEFAULT is the main look of requesters, no matter what title it has or on which screen it's displayed. SCREEN defines the main look for all requesters opened on a selected screen (read the **edit menu** section for details). Both DEFAULT and SCREEN can have some "children", I mean the SUBREQUESTER or SUBPROGRAM. SUBREQUESTER defines the look of a requester specified by it's title, etc. If it's a child of DEFAULT it'll be always displayed with it's settings. If it's a child of SCREEN it'll be displayed with it's settings only when displayed on a "parent" screen. Same goes to SUBPROGRAM. NOTE: if you don't own a GFX board you're of course forced to use screens with different number of planes and sometimes strange palletes - in this case it's better not to define global look for some requesters ("children" of DEFAULT), but to define the look for them as "children" of mostly used screens As you know children always inherit parent attributes. Same goes to RAPrefsMUI. If you edit the DEFAULT or SCREEN settings after "Use" all settings of their children will be updated to "parent" settings, according to "Inherit settings" from menu.

1.9 Main window

1.0 Main menu: Program Prefs

- * don't change Bitmap allocation unless you really need to
- * Gfx path is very useful when defining new structures

1.1 Main menu: Program Prefs: Sound port

From version 1.2 ReqAttack can play samples by some external programs. A supplied RASoundDaemon will play your samples via datatypes. To hear samples the external program has to be launched, and it's port name has to be supplied here. (for eg. RASoundDaemon)

2.0 Prefs menu: Buttons

Edit: opens the
 edit menu
 and let's you change selected entry's
 settings.

Add screen: add's a new screen to the list.

Remove: removes selected entry, if entry is a SCREEN or DEFAULT it'll remove all it's children (DEFAULT can't be removed).

Add requester: add's a new requester to the list as a child of selected father (or child of selected child's father).

Add program: add's a new program to the list as a child of selected father (or child of selected child's father).

Preview: opens a preview requester with the settings of selected entry.

3.0 Titlebar menu

Project: Load...: loads ReqAttack prefs file generated by RAPrefsMUI. This program is able to read only self generated prefs files because of the level of possibilities of prefs file layout and not flexible load engine :(

Project: Save as...: saves ReqAttack prefs file as a normal RAPrefsMUI

generated file (will work as a reqattack.prefs file).

Edit: Mark: marks selected entry. This function is similar to clipboard "copy ←",

but it only stores the selected entry's POSITION from list. So if you remove/change it's position you won't get it's prefs with Paste.

Edit: Paste: copies prefs from previously marked position to selected position ←
.

Edit: Entry's age: moves selected entry (also with it's children) up (younger) or down (older). To understand this take a look at this example
.

1.10 Entry's age example

You've defined something like this: ...SUBREQ | NUMBUTTONS>=2SUBREQ | TITLE="#?About#?"... Most "About" requesters have only one button, but if you'r program displays a requester with title "#?about#?" and it has 2 or more buttons, you'll get the 1st SUBREQ requester's look. To avoid such a problems just make 2nd SUBREQ "younger" (or the 1st one "older"). If you still don't understand take a look at STOP command in ReqAttack Guide.

1.11 Edit window

1.0 Environment section: Specification

Here you should define how your screen, subrequester or subprogram should be recognised by ReqAttack. We do this with some special keywords (look at the popup menu on the right side of string gadget) which are essential for every type of entry.

1.1 Keywords and templates

All keywords MUST be upper case; don't try to use keywords in strings (for example TASKNAME = "someNAME"); strings and all values with commas have to be enclosed with "".

Types of values:

string - normal string (it's possible to use #? joker)
value - decimal value
key - special decimal key (used only with ASPECT)

Types of compare operations:

= - only = is possible
<>= - possible operations are = < > <> <= >=

- * please note that when you use commas the there's only one compare operation possible and it's = !
- * <string,string,...> or <value,value,...> means that only one value is necessary, but it's also possible to add more string/values separated with commas
- * strings/values separated with commas will be threated like:
IF ((KEYWORD = value) OR (KEYWORD = value2) OR ...)

- * few keywords in one specification will be treated like:
IF ((KEYWORD CMP value) AND (KEYWORD2 CMP2 value2) AND ...)
- * don't use same keywords in one specification; only the 1st one will be used

1.2 SCREEN specification

Template: STITLE = "<string,string,...>" - titlebar text
 DEPTH <>= <value,value,...> - number of planes
 HEIGHT <>= <value,value,...> - height in pixels
 WIDTH <>= <value,value,...> - width in pixels
 ASPECT = <key> - screen's aspect

ASPECT key's: 1 - 1:1 screens (ex: 320x256, 640x512)
 2 - 2:1 screens (ex: 640x256)
 4 - 4:1 screens (ex: 1280x256)
 -2 - 1:2 screens (ex: 320x512)
 -4 - 1:4 screens (ex: 320x1024)

- * it's very hard to define the Workbench screen with STITLE, so I advice to define DEFAULT to look great on your WB
- * AGA users: it's good to define a screen specified with "DEPTH<=3" with backgrounds, logos and button images disabled

1.3 SUBREQ specification

Template: TITLE = "<string,string,...>" - requester title
 TEXT = "<string,string,...>" - requester text
 NUMBUTTONS <>= <value,value,...> - # of buttons

1.4 SUBPROG specification

Template: TASKNAME = "<string,string,...>"
 PROCNAME = "<string,string,...>"
 PROGNAME = "<string,string,...>"
 PROGNAME2 = "<string,string,...>"

- * programs can be specified only with one keyword!
- * please read ReqAttack Guide for details about keywords meaning

2.0 Requester and button colors

Color types:

RRGGBB value - normal color definition (hexadecimal), possible to set with sliders or by editing the color string
 Pen number - # of color in screen's palette, possible to set with color string
 Ignore - possible with some of entry's, forces ReqAttack not to draw some of lines, shadows, etc.
 def Pens - colors from Draw-Info stucture of a screen on which the requester will appear

- * it's best to use RRGGBB value, but for some screens pen number can be useful, because ReqAttack will not try to lock some of screen pens (this causes problems sometimes (ex: Organiser))

3.0 Logos, backgrounds, button images

Logo types:

depends on buttons - let's you choose different logos for requesters

with one (info), two (ask) or more (multiask) buttons

normal logo - let's you choose one logo for every requester with current specification

no logo - switches logos off

Built-in logos:

\$INFOLOGO - default image for one button requester

\$ASKLOGO - default image for two buttons requester

\$MULTIASKLOGO - default image for three and more buttons requester

Built-in button images:

\$OKIMAGE - default image for leftmost (OK) button

\$CANCELIMAGE - default image for rightmost (Cancel) button

\$THIRDIMAGE - default image for other buttons

- * clearing the background strings will force ReqAttack to draw backgrounds with colors defined in Requester Colors section
- * clearing the logo or button image string equals to default image

4.0 Menu: Button style

Here you can use one of default ReqAttack button styles. Switching on one of them will set button colors to the predefined settings of selected button style.

4.1 Menu: Inherit changes

When you change the settings of a SCREEN or DEFAULT all children of them will inherit their new settings according to the selected Inherit option. By default all SUBREQs and SUBPROGs will inherit all father's settings without logos, button images and sound.

1.12 Important release notes

1.0 Copyrights

- * ReqAttack is ©1999 by Georg Steger (georg.steger@rolmail.net)
ReqAttack is SHAREWARE!!!
- * RAPrefsMUI is ©1999-2000 by Jaca/D-CAPS (jacadcaps@poczta.onet.pl)
- * ReqOFF is ©2000 by Jaca/D-CAPS (jacadcaps@poczta.onet.pl)
- * ReqAttackPrefs is ©1999 by Jaca/D-CAPS (jacadcaps@poczta.onet.pl)
- * RASoundDaemon is ©1999 by Stephan Rupprecht

2.0 BETA BETA BETA

Please note that both ReqAttack and RASoundDaemon are beta versions! I've been testing them for months now and their never hanged, so I suppose it should be safe to use them...

3.0 Programs in this release

- * RAPrefsMUI - version 1.61 - fixed + ReqOFF support
- * ReqAttack - version 1.2b
- * RASoundDaemon - early beta

* ReqOFF - version 0.4b - NEW in this release

1.13 Bugs, limitations, future versions

* ReqAttack 1.0 and 1.1 won't read the prefs files generated by this program - few options used in 'reqattack.prefs' are supported from version 1.2+

Please let me know if you find any bugs in RAPrefsMUI!
I'll try to fix them in future versions.

1.14 Thanks

Thanx to:

Georg Steger {georg.steger@rolmail.net} for supporting my project and some documentation.

Cyran/D-CAPS {cyran@amiga.w.pl} for downloading ReqAttack archives from aminet and his oppinions about ReqAttackPrefs.

Laffick/D-CAPS {unplugged} for betatesting

Greetz to all AUG-Torun

1.15 History

History: (releases)

1.16	07.12.99	!RELEASED ON ACS CD 01/2000!	(Feb 2000)
1.17	18.12.99	!AVAILABLE ON AMINET!	(Jan 2000)
		(util/wb/RAPrefs.lha)	
		!AVAILABLE ON MrDark's HOMEPAGE!	(Jan 2000)
		(amiga.is.o.k.pl)	
		!AVAILABLE ON MY HOMEPAGE!	(Mar 2000)
		(beton.art.pl/jaca)	
		!RELEASED ON ACS CD 03/2000!	(May 2000)
1.28a	22.03.00	!INFO ABOUT DEVELOPER WORK!	(Mar 2000)
		(beton.art.pl/jaca)	
1.60	21.05.00	!AVAILABLE ON AMINET!	(May 2000)
		(util/misc/reqattackupd.lha)	
		!AVAILABLE ON MY HOMEPAGE!	(May 2000)
		(beton.art.pl/jaca)	
1.61	27.05.00	!AVAILABLE ON AMINET!	(May 2000)

1.16 ReqOFF patch

1.0 What's ReqOFF?

ReqOFF is a patch for Intuition's EasyRequestArgs(). It'll install itself as this function and will filter all requesters that your software wants to display. A requester defined in RAPrefsMUI as 'ReqOFFed' will not be showed and the program that called EasyRequestArgs() will receive the button number specified in RAPrefsMUI (it'll think that this button was pressed).

2.0 Installation

It's very simple; just put it in your startup-sequence. Remember that it has to be launched with 'Run >NIL:' and of course AFTER ReqAttack. (If ReqAttack is in the WBStartup drawer you should move it to C: and also run it from startup-sequence.)

2.1 Quitting

It's possible to remove ReqOFF with Break command or Ctrl C. Please note that ReqAttack shouldn't be turned off before ReqOFF! Always quit ReqOFF 1st!

2.2 Installation: RAPrefsMUI configuration

To turn off a requester switch the ReqOFF function in SUBREQ's environment and specify the button number. The leftmost (OK) button is 1 and the rightmost (CANCEL) is 0, middle buttons are 2,3,4... (numbers grow from left to right of course).

3.0 Limitations

- * ReqOFF only checks the requester's title, so the SUBREQ needs to be defined by TITLE (it can also be defined by TEXT or NUMBUTTONS but ReqOFF won't notice that)
- * ReqOFF doesn't care about the SCREEN
- * ReqOFF will not notice the change of reqattack.prefs, it has to be restarted :(
- * don't run ReqOFF twice (ReqOFF won't notice that :((

3.1 What will change in future?

- * ReqOFF will look at the SCREEN's STITLE and SUBREQ's TEXT
- * support for all Intuition's requester functions
- * send me your ideas!

4.0 Legal stuff

ReqOFF is FREEWARE! Spread it only in this archive!
ReqOFF is ©2000 by Jaca/Dreamolers-Clever Amigants Polish Society

(jacadcaps@poczta.onet.pl)
