

DIRGC

COLLABORATORS

	<i>TITLE :</i> DIRGC		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		February 12, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	DIRGC	1
1.1	main	1
1.2	installation	2
1.3	guis	2
1.4	config	4
1.5	tips	5
1.6	made	5
1.7	commands	6
1.8	other	9
1.9	author	11

Chapter 1

DIRGC

1.1 main

**** DIR.gc Filemanager ****

by D.C. Keletsekis - January 1997
<http://users.hol.gr/~dck/gcmain.htm>

Dir.gc is a nice little file manager made with the Gui4Cli language. It's fast, it's small, it's easy and it features pop up windows for executing it's commands, so that the mouse travelling distance is as short as possible.

Installation
Copy it anywhere..

How to use it
The gui explained..

Configuration
How to configure it..

Tricks & Tips
Usefull information..

Expand it!
Hack it, rip it, change it..

Gui4Cli Commands
All the commands..

Other Commands
Rep and Wrap..

Author
That's me! :)

** Enjoy.. **

1.2 installation

1. Install Gui4Cli :

To do this, just copy the files "Gui", "Gui4Cli", "Wrap" and "rep"
(which are in the c directory) to your c: directory.

IMPORTANT : Included here is a newer version of c:Gui to the one
available in Gui4Cli v2.2 main archive on Aminet (dev/gui).
This version uses the WBPPath routine written by Ralph Babel which
makes sure that Gui4Cli inherits the defaults paths of your system
even if launched from the Workbench (which normally doesn't pass
on your paths to the programs it launches). Thanks to M. van Elst
for pointing it out to me. Details of the WBPPath routine are
included herein.

2. Install the file manager :

Copy the "DIR" directory anywhere in your hard disk.

3. Run it.. :

From the WB, run it by clicking on it's icon.

From the shell, type > run gui dh0:dir/dir.gc
(or where ever you have placed the directory)

NOTE : Be sure to
 Configure
 it before you start
clicking on files, or you'll probably get a lot of errors.

1.3 guis

Some boring explanations :

----- The buttons :

P = Parent dir
R = Root dir
D = Device list
A = Select All
N = Select None

> = Open the other half of the window and display the other LV also.
x = Switch left <-> right listviews.
= = Set the other LV to the same dir as the source LV.

R = Reload current directory
< = Shift right
> = Shift left

Note that there is no set "source" or "destination". The Source is always the ListView which was most recently clicked upon. This may be confusing at first but I'm trying to be intuitive here, so give me a break..

The pop-up gui which opens when you click the right mouse button will show you which LV is source and which destination.

----- The Guis :

This is just to explain some of the peculiarities of the guis that pop-up.

Calculator : (File dir/dir.calc)

This is a full calculator with a ticker tape etc, which uses ARexx to calculate the result. Click on the RMB and the window will open up and provide you with a ticker tape display. Click on any line in the listview, and the main calc window will be updated with the value on that line.

This GUI is actually an independent gui and can be operated by itself if needed, or attached to any other guis you make.

Avail : (File dir/dir.avail)

This is a confusing GUI. It is meant to show you how much memory is used up by various programs you run. However, it saves the memory display into a little file in ram, so it does not always show it correctly (The difference is about 8 bytes).

If you want a true memory count, click on "flush" and then on the "avail" and "reset" buttons repeatedly, until you get a 0 in both the displays. Then run the program you want to check and then click on "avail". The memory used should be displayed correctly.

It helps a lot if you know what you're doing..

Lock : (File dir/dir.lock)

This is a lock. When clicked upon, a full screen borderless window will appear, with a text-in gadget in it.

Input the pass code you want and press enter.

The text-in gadget will go blank and the window will stay there and not go away unless and until you re-enter the pass code you entered the first time.

PPShow : (file dir/dir.ppshow)

Hit the PPShow button and you'll get a gui which will play all selected picture/anim files in order, with the settings you have given in the gui.

You MUST have the PPShow program (by Nico Francois) in your c: dir.

Although this gui is only for the PPShow viewer, with a little tampering of the script file you can change it for any other player you prefer. The options (Speed, resolution etc) will be similar, so the changes will be few.

GfxCon : (File dir/dir.gfxcon)

To run this gui you need the GfxCon picture converter from Aminet. Once you get it, configure dir.gc again, assigning the DEF.GFXCON variable to the name&path of the program and you're off..

1.4 config

Click on the right mouse button, then on "More..".

Then click on "Config" and you'll get a gui where the default players for each type of file will be shown. Most of their uses are pretty obvious, but listed hereunder for purely historical reasons..

```
DEF.EDITOR      = full path of your favourite editor
DEF.MULTIVIEW   = full path of your Multiview prog.
DEF.GUIDE       = The full path of your amigaguide program
DEF.HEX         = a binary file viewer such as "FileX" from Aminet.

DEF.IIIBM      = a normal amiga IFF picture viewer
DEF.ANIM       = an ANIM player (note - all these could be the same)
DEF.GIF        = a GIF player
DEF.JPG        = a viewer specifically for JPGs, such as FastJpg
DEF.GFX        = a general anim/picture viewer, such as "ppshow"
DEF.GFXCON     = the full path & name of "GfxCon V1.7"
                 (a program for converting pictures - available on aminet)

DEF.8SVX       = a 8SVX sound player such as "playsound"
DEF.MOD        = a Module Player such as "Proplay2.exe" or hippoplayer

DEF.PP        = for power-packed files
```

You can enter some options also in the command line.
The file name will be appended to the command line shown.

I ripped this gui off the CliMaster file manager. Thanks Graham :)

1.5 tips

These are some things that you should know..

----- Changing the Screen :

Use CONTROL-J to switch to the next available public screen. The Gui together with all the accompanying pop-up guis will go to that screen.

----- Using it with icons :

The window of the file manager is an AppWindow. Drag icons into it and it will change to the directory of the icon dragged. This is also true for the AppIcon that appears when the gui closes.

----- Changing the font :

The font the LV's use is topaz 8, but you can change it to any font you like. Only it should be a monospaced font, otherwise the alignment will not look correct. To change the font, edit the Dir.gc file and change the GadFont command of the 2 ListViews. Go on.. I know you can do it..

----- Setting the window size you want :

The file manager window is resizable. You can resize it to the size you want and then set this size as the default size the manager will always open with, by pressing CONTROL-W and clicking on the "BIG" button. The LVs may not look very good at some sizes. This is because GadTools automatically sizes the LV to show the max No of lines it's able to.

----- DoubleClicking on a file :

In this case, a small gui will open allowing you to decide what to do with the file chosen. The top choice is "Auto". This calls a routine which resides in file dir/dir.View. This routine will check the file type and run it accordingly, using the default player programs you have defined in 'Config'

1.6 made

You know how all programs have file requesters ?...

Well, you can think of this as a requester which has many programs.

Gui4Cli is a very simple GUI constructing language. It's simpler than

AmigaDos and with a little effort you can do a hell of a lot of stuff..

It is a very good idea to look at the scripts. You will not understand most of it, but you will get a general feeling of what's going on.

HOW TO EXPAND IT :

When you click the right mouse button, then "More..", you will see a button called "User..". Hitting this button will load and display an empty pop-up button window, like the normal ones.

You can assign any other GUIs you want to these buttons.

The file of this GUI is dir/dir.user

Load it and take a look..

It's best not to use/alter the other buttons, as I'm going to be expanding the file manager, and next versions may overwrite your changes.

1.7 commands

This is a simple listing of all the commands available in Gui4Cli V2.2 For a full explanation of the commands, plus demos, get the full archive of Gui4Cli from Aminet at dev/gui/Gui4Cli.lha

===== GLOBAL COMMANDS =====

```
WinBig          L T W H Title
WinSmall        L T W H
WinType         MASK (Close|Drag|Zoom|Depth|Borderless|Backdrop|RIGHT|BOTTOM)
WinOut          ConsoleSpecification
WinOnWin        GuiName LeftOffset TopOffset
WinOnMouse      LeftOffset TopOffset
Screen          PublicScreenName
WinFont         FontName Size UL|BD|IT(Mask)
WinBackground   SOLID|PATTERN|ICON APen|IconName BPen
UseTopaz
NoFontSense
```

===== GRAPHICS COMMANDS =====

```
BOX            L T W H IN|OUT BUTTON|RIDGE|ICONDROP
CTEXT          L T Text FontName size FGpen BGpen MASK(Underline|Bold|Italics)
LINE           L T L T ColorNo
SQUARE         L T W H ColorNo FILL|NOFILL
CIRCLE         centerL centerT xradius yradius ColorNo FILL|NOFILL
```

```
===== EVENTS =====
```

```

xButton      L T W H Title
xCheckBox    L T W H Title Variable OnText OffText ON|OFF
xVSlider     L T W H Title Variable Min Max Current ShowStr
xHSlider     L T W H Title Variable Min Max Current ShowStr
xTextIn     L T W H Title Variable StartingText Bufflength
xCycler      L T W H Title Variable
xListview    L T W H Title Variable FromFile Offset NUM|TXT|DIR|MULTI
xRadio       L T W H Variable Spacing
xMemu        Menu Item SubItem Shortcut
xIcon        L T IconName (no .info)
xAppMenu     AppMenuName Variable ONOFF
xAppIcon     L T IconName Title Variable ON|OFF
xAppWindow   Variable

xOnKey       Letter|KeyValue
xRoutine     RoutineName
xLVDirHook   HookID
xOnReturn    LaunchID
xOnJump      NewScreenName
Text         L T W H Text Length BOX|NOBOX

Other Events  xOnLoad, xOnOpen, xOnClose, xOnQuit
              xOnRMB, xOnActive, xOnInactive, xBefore, xAfter
              xOnDoubleClick, xOnFail, xOnReload

```

```
----- GADGET MODIFIERS -----
```

Gadget Modifiers :

```

GadID        IDNumber
GadFont      FontName FontSize MASK(Underline|Bold|Italics)
GadTitle     ABOVE|BELOW|LEFT|RIGHT|HIGH
GadKey       Letter (or #ASCII value)
GadTxt       LEFT|CENTER|RIGHT
LVColors     Foregnd Backgnd SelectedBgnd Dirs
LVDirHook    HookID

```

```
===== EVENT COMMANDS =====
```

-----> Controlling Gadgets :

```

SetGad       GuiFile GadID ON|OFF|SHOW|HIDE (Arexx capable)
Update       GuiFile GadID Value (Arexx capable)
ChangeArg    GuiFile GadID ArgNumber NewValue
ChangeGad    GuiFile GadID L T W H Title
ReDraw       GuiFile
ChangeIcon   GuiFile GadID L T NewIconName

```

-----> Control Statements :

```

If/Else/Endif          Argument Operator Argument
IfExists/Else/EndIf   SystemItem Name
While/EndWhile         Argument Operator Argument
Mark/Goto              MarkName
Gosub/Return          GuiName RoutineName (ARexx capable)
DoCase/Case/Break/EndCase (DoCase) Argument - (Case) Operator Argument
Stop

```

----- All Commands below this line are ARexx capable -----

Quit

-----> Commands which set the \$\$RETCODE :

```

Run, CLI      CommandLine
Launch       LaunchID CommandLine
SendRexx     PortName  CommandLine
Wait        SystemItem Name Timeout
Delete       File/Dir          (with wild characters)
Copy        File/Dir Destination (with wild characters)
MakeDir      DirName
assign      Device: Path|REMOVE
rename      OldName NewName
FailAt      ErrorNumber

```

```

Action      Action  Source Destination
LVAction    Action  GuiFile GadID  Destination

```

-----> Handling GUIs :

```

GuiLoad      FullPath/GuiName
GuiOpen      GuiName
GuiClose     GuiName
GuiQuit      GuiName
GuiRename    GuiName NewName
Status

```

-----> Handling Variables :

```

SetVar       Variable String
DelVar       Variable (May use wildcard eg. var#?)
AppVar       Variable Text
CutVar       SourceVar CUT|COPY CHAR|WORD|LINE Amount DestinationVar
Counter      Variable INC|DEC Amount
Append       File String
Extract      Variable Root|File|Path|Guipath|Clean|Unquote|Ext
              Upper|Lower ToVariable
JoinFile     Path File ToVariable

```

-----> ListView Commands :

```

LVChange     GuiFile GadID NewFromFile
LVSort       GuiFile GadID
LVFind       GuiFile GadID String
LVAdd        GuiFile GadID String
LVDel        GuiFile GadID LineNumber

```

```

LVClear      GuiFile GadID
LVSave       GuiFile GadID FileName
LVGet        GuiFile GadID LineNumber Variable
LVPut        GuiFile GadID LineNumber NewText
LVMove       GuiFile GadID +-Offset

```

-----> Dir/Multi Listview Commands :

```

LVDir        GuiFile GadID Parent|Root|Disks|All|None|#DirName
              Refresh|Norefresh
LVMulti      GuiName GadID First|Next|On|Off

```

-----> Various Commands :

```

SetScreen    GuiFile ScreenName
GuiScreen    GuiFile FRONT|BACK
GuiWindow    GuiFile ON|BIG|SMALL|FRONT|BACK|WAIT|RESUME
SetWinTitle  GuiFile NewTitle
ReqFile      L T W H Title SAVE|LOAD|MULTI|DIR Variable DirName
CD           NewDirectoryName
Delay        Ticks
EZReq        Text Choices Variable
Say          Text
Set          [parameter] [value]
SetStack     StackSize
MakeScreen   ScreenName Depth Title
KillScreen   ScreenName
SetColor     GuiFile ColorNumber R G B
TTGet        FullPath/IconName (without .info)

```

ACTIONS : COPY COPYNEW MOVE DELETE PROTECT SIZE

SYSTEM : PORT SCREEN FILE DIRECTORY GUI WINDOW VARIABLE
 DEVICE VOLUME ASSIGN DOS

1.8 other

CLI Commands :

----- c:REP

This is a small, fast CLI command for replacing strings in files.
 It is pure and can be made resident.

Use : Rep FromFile/A, String/A, ToString/A, ToFile, I=Insensitive/S

where :

```

FromFile    - is the file you want to convert.
String       - The string you want to replace
ToString     - What you want to replace it with
ToFile       - (Optional) the name of the output file. If this is not
              given, then the INPUT FILE WILL BE OVERWRITEN!
Insensitive - (Optional) Make the search case-insensitive

```

example :

```
rep df0:MyFile ThisString ThatString Insensitive
- will replace all occurrences of "ThisString" (or THISstring, or
  ThisStRiNg etc) with "ThatString" and save it over the original
  file.
```

----- c:WRAP

This is a small CLI command for rewrapping text files.
It is pure and can be made resident (Resident c:wrap pure add)

Usage :

Wrap FILE/A,NEWFILE,L=LENGTH/K/N,TAB/K/N,P=PARA/K/N,STRIPCR/S,ADDCR/S

FILE = Must be given. It's the file which you want to rewrap and it must be a text file!

WARNING - if you re-wrap a binary or a PowerPacked or XPK etc file, the file will become trash!

NEWFILE = This is the name of the output file. If it is not given, then the input FILE will be overwritten!

LENGTH = The new line length you want. If this is not given, the default is 1000 characters, which results in leaving the line length same (unless you have lines over 1000 characters long). If the Length given is too small to fit a single word, then the word will be split. Max length is 1000 characters.

TAB = If this option is given, then any tabs found will be dealt with in accordance with the number of spaces specified, and be converted to spaces. Maximum tab size is 80 characters.

Warning If TAB size is not declared, all tabs will be deleted. This is not as bad as you may imagine, since Re-wrapping files with a lot of tabs looks ugly anyway..

PARA = This option is weird but usefull. It lets you define the number of NEWLINE characters that Wrap will read before putting 1 NEWLINE character. Most text documents have a NEWLINE character at the end of every line, and 2 NEWLINES to denote a paragraph.

So if you put PARA=2 you will get much better formatted text because the single NEWLINE characters will be ignored. This option also allows you to re-wrap files to a longer line length than the original file.

The default is PARA=1, i.e. leave them un-altered. Maximum is 30 newline characters.

STRIPCR = Strips those PC Carriage Return charaters.

ADDCR = Adds Carriage returns after every NEWLINE character, so

you can tell PC users what you really think..

By declaring both STRIPCR *and* ADDCR you could ReWrap a PC file and keep it in PC format.

Example :

```
Wrap MyFile MyOutputFile LENGTH=60 TAB=5 PARA=2 STRIPCR
```

1.9 author

Author :

Dimitri C. Keletsekis
14 King George str.
Athens 10674
Greece

E-Mail : dck@prometheus.hol.gr
WWW Page : <http://users.hol.gr/~dck>

This file manager is free-ware, but remains the copyright of the author. Absolutely no guarantees are given or implied as to it's actions or any consequences arising from it's use. Use it at your own risk.

Gui4Cli (which is also written by me) is shareware, but not disabled in any way. Full versions can allways be found on aminet under dev/gui/Gui4Cli.lha or at my web page.

Rep & Wrap are FreeWare
