

Gad

Mike Archer

COLLABORATORS

	<i>TITLE :</i> Gad		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Mike Archer	November 29, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Gad	1
1.1	main	1
1.2	Gad Grid - Main Instructions	1
1.3	Gad Grid - Running from WorkBench	2
1.4	Gad Grid - Running GadGrid from workbench	2
1.5	projecticon	2
1.6	shiftsel	3
1.7	wbstartup	3
1.8	Gad Grid - A Handy Tip	3
1.9	Gad Grid - Setting the Tool Types	4
1.10	Gad Grid - The Default tool type.	4
1.11	Gad Grid - The WindowX tool type.	5
1.12	Gad Grid - The WindowY tool type.	5
1.13	Gad Grid - The HideX tool type.	5
1.14	Gad Grid - The HideY tool type.	6
1.15	Gad Grid - The FontName tool type.	7
1.16	Gad Grid - The FontSize tool type.	7
1.17	Gad Grid - The Columns tool type.	7
1.18	Gad Grid - The WindowWidth tool type.	8
1.19	Gad Grid - The Console tool type.	8
1.20	Gad Grid - Running from CLI	8
1.21	Gad Grid - The FILE parameter	9
1.22	Gad Grid - The FONT & SIZE parameters	9
1.23	Gad Grid - The COLS parameter	10
1.24	Gad Grid - The WIDTH parameter	10
1.25	Gad Grid - The CONSOLE parameter	11
1.26	Defining a console window	11
1.27	Gad Grid - The TOP & BOTTOM parameters	12
1.28	Gad Grid - The LEFT & RIGHT parameters	12
1.29	Gad Grid - A really cool tip for CLI users	12

1.30	requi	13
1.31	Gad Grid - Hiding the Gadget Grid	14
1.32	Gad Grid - Defining the Grid-Definition file	14
1.33	Example of a basic Grid-Definition file	15
1.34	Gad Grid - Introduction	15
1.35	Gad Grid - Disclaimer	16
1.36	Gad Grid - Features	16
1.37	Gad Grid - Installation	17
1.38	Gad Grid - Distribution	17
1.39	Gad Grid - About the Author	17
1.40	Gad Grid - Known bugs	17
1.41	Gad Grid - Trouble Shooting	18
1.42	Solution	18
1.43	Gad Grid - Version History	18
1.44	Gad Grid - Future Plans	19
1.45	Gad Grid - Problems with power-packer	20

Chapter 1

Gad

1.1 main

Gad Grid V1.1

©1997 by Mike Archer

CONTENTS

INTRODUCTION

DISCLAIMER

FEATURES

INSTALLATION

REQUIREMENTS

DISTRIBUTION

ABOUT AUTHOR

HOW TO USE GAD GRID

TROUBLE SHOOTING

KNOWN BUGS

VERSION HISTORY

FUTURE ENHANCEMENTS

WARNING

Do not crunch GadGrid with Power-Packer!!

If GadGrid is crunched with Power-Packer it will crash!!

Why ?

1.2 Gad Grid - Main Instructions

Defining the Grid-Definition file

Running GadGrid from WorkBench

Running GadGrid from CLI

Hiding the GadGrid window

1.3 Gad Grid - Running from WorkBench

You can run GadGrid from WorkBench using a number of methods, these are...

* Double-Click on GadGrid's icon, doing this will load the Grid-Definition file that is named as "Default" in the Tool-Types of the GadGrid icon.

[More Details](#)

* Set up a "Project" icon for a Grid-Definition file and make GadGrid the default tool for that icon.

[More Details](#)

* Click on GadGrid's icon, hold down the shift key and double-click on any Grid-Definition file.

[More Details](#)

* Launch GadGrid from your WbStartUp drawer.

[More Details](#)

With any of the above methods of running GadGrid, if the Grid-Definition file has an icon then any ToolTypes for this icon will overule the ToolTypes set in GadGrid's own icon.

[Setting the ToolTypes](#)

1.4 Gad Grid - Running GadGrid from workbench

If you run GadGrid by double clicking on it's WorkBench icon then GadGrid will load the Grid-Definition file named by the **Default** ToolType.

If there is no "Default" ToolType set then GadGrid will attempt to load a Grid-Definition file called "Grid.Def" from the same drawer as the GadGrid program. If this can't be found then GadGrid will fail.

1.5 projecticon

Once you have created a Grid-Definition file you can give it a "Project" icon.

Set the "Default Tool" in the project icon to GadGrid and now you can run GadGrid with that Grid-Definition by simply double-clicking on the Project Icon.

You may specify all the same ToolTypes in this Project Icon as you can in GadGrid's program icon. (Except the "Default" ToolType)

1.6 shiftsel

If you have created a Grid-Definition file, but not given it a "Project" Icon you may run GadGrid with that Grid-Definition file by clicking once on the GadGrid icon, then holding down the SHIFT key double click on the Grid-Definition file you wish GadGrid to load.

NOTE : Although WorkBench allows you to select more than one file when using this method, GadGrid will only use the first filename passed to it from WorkBench.

1.7 wbstartup

If you want a GadGrid to be run automatically when your machine boots-up then you may do one of two things...

- * Place the GadGrid program into you WBStartup drawer and then set the "Default" ToolType to specify the Grid-Definition file to load.
- * Place the Grid-Definition file (with a Project Icon) into the WBStartup drawer.

With either method you should add the "DoNotWait" ToolType to the file you have placed in the WBStartup drawer. This ToolType tells WorkBench not to wait for GadGrid to finish running.

A Handy Tip for using GadGrid on your WorkBench screen

1.8 Gad Grid - A Handy Tip

If you are going to have GadGrid permanently installed on you WorkBench then you may take advantage of the AutoScroll setting in the ScreenMode preferences program. You can then have a WorkBench screen wider than the Visible area and hide GadGrid out of normal view.

EXAMPLE

- * Run the ScreenMode program in the Preferences drawer.
- * Click on the "Default" Check-Mark box next to the "Width" box so there is no tick in it.
- * You can now enter your own width for the Workbench screen in the "Width" box. Add 100 to the width already displayed.
- * Click on the "AutoScroll" Check-Mark box so that a Tick is displayed.
- * Save these settings.
- * Place your Grid-Definition file in the WBStartup Drawer, and set the following ToolTypes...

WindowX=RIGHT

WindowY=TOP

HideX=SCREEN|RIGHT

HideY=SCREEN|TOP

WindowWidth=100

Columns=1

Now when your Workbench boots up in future you will have a Workbench screen that looked just the same as before, but move the mouse pointer to the right edge of the screen and on scrolls a column of gadgets to launch your favourite programs!

You could also leave the width the same, but add to the height of your WorkBench screen so the GadGrid window is hidden of the bottom of your WorkBench Screen. To do this set the following ToolTypes...

WindowY=BOTTOM

HideX=WINDOW|TOP

HideY=WINDOW|LEFT

How much you add to the height of the WorkBench really depends on how many Gadgets you have in the grid and what Font Size you use. You'll just have to experiment a bit!

1.9 Gad Grid - Setting the Tool Types

GadGrid will recognise the following ToolTypes...

Default - Name of the file to load.

WindowX - Horizontal position of the window.

WindowY - Vertical position of the window.

HideX - Horizontal position of the "Hide" window.

HideY - Vertical position of the "Hide" window.

FontName - Name of the font to use.

FontSize - Size of the font to use.

Columns - Number of gadget columns in grid.

WindowWidth - Width of the window.

Console - Name of the console window.

1.10 Gad Grid - The Default tool type.

The "Default" tool type may only be used in GadGrid's program icon, it will be ignored in "Project" icons.

This tool type specifies the name of the Grid-Definition file to load

when the GadGrid program icon is double-clicked.

EG: Default=DH0:DiskTools.Grid

If only a filename is given and no path then GadGrid will look in the drawer that the GadGrid program is located.

The default value for this tool type is "default.grid".

1.11 Gad Grid - The WindowX tool type.

The "WindowX" tool type is used to specify the horizontal position of GadGrid's window. It only has any effect if you have specified your own **Window Width** otherwise the window fills the width of the screen.

The "WindowX" tool type can have one of three values...

- * LEFT - Opens the window against the left hand edge of the screen.
- * MIDDLE - Opens the window in the middle of the screen.
- * RIGHT - Opens the window against the right hand edge of the screen.

EG: WindowX=LEFT

The default value for this tool type is "MIDDLE"

1.12 Gad Grid - The WindowY tool type.

The "WindowY" tool type is used to specify the vertical position of GadGrid's window.

The tool type can have one of three possible values...

- * TOP - Opens the window at the top of the screen.
- * MIDDLE - Opens the window in the middle of the screen.
- * BOTTOM - Opens the window at the bottom of the screen.

EG: WindowY=TOP

The default value for this tool type is "BOTTOM"

1.13 Gad Grid - The HideX tool type.

The "HideX" tool type is used to specify the horizontal position of the small "Hide" window that GadGrid opens when you choose to Hide the main GadGrid window.

The "HideX" tool type expects two values separated by the vertical bar character (|).

This first of these can be one of two values...

- * SCREEN - The Hide Window is positioned relative to the screen edges.
 - * WINDOW - The Hide Window is positioned relative to the window edges.
-

When the first value is SCREEN, the second value can be one of...

- * LEFT - The Hide window will be opened against the left hand edge of the screen.
- * MIDDLE - The Hide window will be opened in the middle of the screen.
- * RIGHT - The Hide window will be opened against the right hand edge of the screen.

EG: HideX=SCREEN|LEFT

When the first value is WINDOW, the second value can be...

- * LEFT - The hide window will be opened so it's left edge is in the same position as the left edge of GadGrid's main window.
- * MIDDLE - The hide window will be positioned in the middle of GadGrid's main window.
- * RIGHT - The hide window will be opened so it's right edge is in the same position as the right edge of GadGrid's main window.

EG: HideX=WINDOW|RIGHT

The default value for this tool type is "WINDOW|LEFT"

1.14 Gad Grid - The HideY tool type.

The "HideY" tool type is used to specify the vertical position of the small "Hide" window that GadGrid opens when you choose to Hide the main GadGrid window.

The "HideY" tool type expects two values separated by the vertical bar character (|).

This first of these can be one of two values...

- * SCREEN - The Hide Window is positioned relative to the screen edges.
- * WINDOW - The Hide Window is positioned relative to the window edges.

When the first value is SCREEN, the second value can be one of...

- * TOP - Open the hide window at the top of the screen.
- * MIDDLE - Open the hide window in the middle of the screen.
- * BOTTOM - Open the hide window at the bottom of the screen.

EG: HideY=SCREEN|MIDDLE

When the first value is WINDOW, the second value can be...

- * TOP - Open the hide window so it's top edge is in the same position as the top edge of GadGrid's main window.
- * MIDDLE - Open the hide window in the middle of GadGrid's main window.
- * BOTTOM - Open the hide window so it's bottom edge is in the same position as the bottom edge of GadGrid's main window.

EG: HideY=WINDOW|BOTTOM

The default value for this tool type is "WINDOW|TOP"

1.15 Gad Grid - The FontName tool type.

The "FontName" tool type allows you to change the font that GadGrid uses for the text in the gadgets.

You should also specify the new font size using the **FontSize** tool type.

EG: FontName=diamond.font

The default value for this tool type is "topaz.font"

NOTE : To use the "FontName" Tool Type you must have the diskfont library installed on your system, otherwise the FontName tool type will be ignored.

1.16 Gad Grid - The FontSize tool type.

The "FontSize" tool type allows you to change the size of the font that GadGrid uses for the text in the gadgets.

If you change the font size then GadGrid will make the gadget boxes tall enough to accommodate the new font height and increase the window height to allow for this too.

EG: FontSize=22

The default value for this tool type is "8".

@ENDNODE

1.17 Gad Grid - The Columns tool type.

The "Columns" tool type allows you to specify how many columns of gadgets there will be in the grid.

GadGrid will adjust the width of the gadget boxes to fit the window.

For example if the window was 640 pixels wide and "Columns=10" was used then the gadgets would be 64 pixels wide, but if "Columns=2" was used then the gadgets would be 320 pixels wide.

The lower limit for the "Columns" tool type is 1, there is no upper limit to how many columns of gadgets you may have, but create too many and the gadget boxes will get too narrow to fit any text in!

EG: Columns=5

The default value for this tool type is "10".

1.18 Gad Grid - The WindowWidth tool type.

The "WindowWidth" tool type tells GadGrid exactly how wide the window it opens should be.

There are no limits on the width that you specify, but if the window is too wide to fit on your workbench screen, or too narrow to fit anything in then GadGrid will fail.

If no "WindowWidth" tool type is found, rather than use a default value, GadGrid will check how wide your WorkBench screen is and open a window the full width of your WorkBench screen.

NOTE: There is no facility to specify the height of the window, this is because the height of the window is determined by how tall the gadget boxes are and how many rows of gadgets are needed to fit on all the gadgets specified in the Grid-Definition file.

1.19 Gad Grid - The Console tool type.

When GadGrid launches a script of commands, it first opens a console window for the commands to send any output to. You may specify a different console window with the "Console" tool type.

EG: Console=CON:50/50/200/50/Output/AUTO

The default is "CON:0/0/640/100/GadGrid/CLOSE/WAIT/AUTO"

Although it is possible to specify the name of a file instead of a console window, because GadGrid launches the scripts it runs as background tasks it could be possible that you could launch another script before the last one has finished running. In such a case, if you had sent output to a file both the scripts would be trying to write to the same file. IE: Problems!)

If you don't want any output from the commands to be displayed then you may use a tool type of "Console=NIL:".

How do I define a console window then ?

1.20 Gad Grid - Running from CLI

Older Amigas

If you have an old Amiga with version 1.x of the operating system then the support that GadGrid provides when running from the CLI is very basic.

You may only specify one parameter when running GadGrid, that is the name of the Grid-Definition file to load. All is not lost though, because if the Grid-Definition file has a "Project" Icon then the Tool-Types set in it will be used by GadGrid when running from CLI as well as WorkBench.

Newer Amigas

For anyone with an Amiga that has AmigaDos 2.0 or higher the good news is that alot more can be done from CLI. Here is a list of parameters that you may specify when running GadGrid from CLI.

FILE - The name of the Grid-Definition file to use

FONT - The name of the font to load.

SIZE - The size of the font to load.

COLS - The number of gadget columns to create.

WIDTH - The width of the Gadget Grid window.

CONSOLE - The name of the Console output.

TOP - Open window at screen top.

BOTTOM - Open window at screen bottom.

LEFT - Open window at screen left.

RIGHT - Open window at screen right.

When run from CLI if the Grid-Definition file has a "Project" Icon then GadGrid will take the settings from the Tool-Types in that icon.

Any parameters given in CLI command will override the parameters in the Tool-Types.

A Really cool tip for CLI users

1.21 Gad Grid - The FILE parameter

The FILE parameter specifies the name of the Grid-Definition file to load.

It is the only parameter that doesn't require a keyword before it.

Any text that is not recognised as another parameter will assumed to be the name of the Grid-Definition file.

1.22 Gad Grid - The FONT & SIZE parameters

You may change the font that GadGrid uses for the text inside the gadget boxes to any font you wish. GadGrid will recalculate the height of the gadget boxes to fit the new font.

The FONT parameter specifies the name of the font to use, and the SIZE parameter specifies the size of the font to use.

EG: GadGrid grid.def FONT diamond.font SIZE 12

You can just use the FONT or SIZE parameter on their own, they do not always have to be given together. In such cases the parameter you did not specify will be taken from the Grid-Definition's icon file Tool-Types. (Or the default if not specified by ToolTypes either).

The default is "FONT topaz.font SIZE 8"

NOTE : To change the font you must have the diskfont.library in the LIBS: directory. If you don't GadGrid won't fail, but it will ignore any instructions to change the font.

1.23 Gad Grid - The COLS parameter

You may tell GadGrid how many columns of gadgets there will be in the window using the COLS parameter.

You must always include the keyword COLS before the actual number.

EG: GadGrid grid.def COLS 5

GadGrid will adjust the width of the gadget boxes to fit the window.

For example if the window was 640 pixels wide and "COLS 10" was used then the gadgets would be 64 pixels wide, but if "COLS 2" was used then the gadgets would be 320 pixels wide.

The lower limit for the COLS parameter is 1, there is no upper limit to how many columns of gadgets you may have, but create too many and the gadget boxes will get too narrow to fit any text in!

If you do not specify a COLS parameter the default is "COLS 10".

1.24 Gad Grid - The WIDTH parameter

You may tell GadGrid exactly how wide the window it opens should be by using the WIDTH parameter.

You must always include the keyword WIDTH before the actual number.

EG: GadGrid grid.def WIDTH 320

There are no limits on the width that you specify, but if the window is too wide to fit on your workbench screen, or too narrow to fit anything in then GadGrid will fail.

The no width is given for the window then GadGrid will open a window the same width as your WorkBench screen. (GadGrid does not assume that this is 640 pixels, many other menu programs do!)

NOTE: There is no facility to specify the height of the window, this is because the height of the window is determined by how tall the gadget boxes are and how many rows of gadgets are needed to fit on all the gadgets specified in the Grid-Definition file.

1.25 Gad Grid - The CONSOLE parameter

When GadGrid launches a script of commands, it first opens a console window for the commands to send any output to. You may specify a different console window with the CONSOLE parameter.

EG: GadGrid grid.def CONSOLE CON:50/50/200/50/Output/AUTO

The default is "CON:0/0/640/100/GadGrid/CLOSE/WAIT/AUTO"

Although it is possible to specify the name of a file instead of a console window, because GadGrid launches the scripts it runs as background tasks it could be possible that you could launch another script before the last one has finished running. In such a case, if you had sent output to a file both the scripts would be trying to write to the same file. IE: Problems!)

NOTE: Although it is possible from Workbench, if from CLI you give a name of "NIL:" output will not be sent to the "NIL:" device, but instead will be output in the CLI window that GadGrid was run from. This seems to be a problem with the EXECUTE command.

Try creating a small script that produces some output and then typing "EXECUTE >NIL: <script-name>" and execute will not send the output to NIL: as you specified.

As a general rule, do not use "NIL:".

How do I define a console window then ?

1.26 Defining a console window

A console window can be specified as follows..

CON:x-pos/y-pos/width/height/text/args

x-pos - Specifies the horizontal position of the window.

y-pos - Specifies the vertical position of the window.

width - Specifies the width of the window.

height - Specifies the height of the window.

text - Specifies the text to be put in the window's title bar.

args - Under AmigaDos 2.0 upwards extra arguments may be specified after the 'text' parameter. Each of these arguments should be seperated by a slash "/" character. The arguments allowed are as follows...

AUTO - If the auto parameter is given then the console window will not be opened unless any output is written to it.

CLOSE - Adds a Close Gadget to the console window.

WAIT - Holds back closing the window until the user clicks on the Close Gadget. (or presses CONTROL & BACKSLASH)

1.27 Gad Grid - The TOP & BOTTOM parameters

You may specify where abouts on your WorkBench screen GadGrid will open it's window.

The TOP parameter tells GadGrid to open it at the Top of the screen.

Suprisingly, BOTTOM tells GadGrid to open it at the Bottom.

If you do not specify TOP or BOTTOM then GadGrid will open it in the centre of the screen. (Or the position specified in the ToolTypes for the Grid-Definition file)

1.28 Gad Grid - The LEFT & RIGHT parameters

If you specified a WIDTH parameter then the GadGrid window will not be the same width as your WorkBench screen. Normally under such circumstances GadGrid will open it's window in the centre of the Screen.

Using the LEFT parameter will tell GadGrid to open it's window at the left hand side of the screen.

The RIGHT parameter similarly tells GadGrid to open it's window on the right hand side.

1.29 Gad Grid - A really cool tip for CLI users

When GadGrid looks at the Grid-Definition file any lines of text before the first Seperator-Line will be ignored.

A good way to put this to use is to create a small script before the first Seperator-Line that will load up GadGrid. Then set the 's' flag and you can now load up a Grid-Definition file by just typing it's name!

For example, I have a Grid-Definition file called "MusicMenu" that looks like this...

```
+-----+
|GadGrid MusicMenu COLS 1 FONT basic.font SIZE 22 |
|Quit |
|#Pro Tracker |
|SYS:MusicTools/ProTracker |
|Etc... |
```

```
+-----+
```

After creating the file I typed in the following CLI command...

PROTECT MusicMenu +s

Now I can simply type in "MusicMenu" to call up this Gadget Grid.

NOTE: You must always put a "Quit" before the first Separator-Line when doing this. Otherwise AmigaDOS will try to execute all the lines in the Grid-Definitions after the Separator-Line too.

1.30 requi

GadGrid requires the following things to run properly...

(1) - EXECUTE command

To run GadGrid must have access to the "EXECUTE" command.

If you have AmigaDos 2 or higher then the execute command is resident in your machines ROM and will always be present.

If you have AmigaDos 1 then you must ensure that your C: directory contains the execute command.

If you are creating a bootable disk that uses GadGrid be sure to copy the EXECUTE command onto the disk, or users with older machines will find GadGrid fails to run properly.

(2) - DISKFONT.LIBRARY

If you wish to change the font used for the Gadget Text from "Topaz" to a font of your choice. Then GadGrid will require the diskfont.library to be present in your LIBS: directory.

If the diskfont.library can't be opened when GadGrid runs it will not fail, but any instructions to change the font will be ignored.

(3) - T:

GadGrid creates a temporary batch file every time a gadget is clicked on. These files are created in the drawer that T: is assigned to.

These temporary files will be deleted automatically by GadGrid as it finishes with them. Because this involves a lot of Reading & Writing to the T: drawer, it is a good idea to have this in the RAM: disk.

The T: drawer will be assigned to RAM: for you already when you boot workbench. If you are creating your own bootable disk that uses GadGrid though, you should ensure that the StartUp-Sequence contains the command "ASSIGN T: RAM:" or similar.

1.31 Gad Grid - Hiding the Gadget Grid

If you click on the CLOSE gadget of GadGrid's window, gadget grid will not exit as you would expect.

Instead GadGrid simply closes the window and opens a tiny window which consists of merely a Title-Bar. This window is known as the "Hide Window".

When you want to use GadGrid again simply click on the Hide Window with both the Left & Right mouse buttons at once. GadGrid will then re-open it's main window containing all the Gadgets.

(GadGrid can also be sent into Hide mode by selecting the "Hide" option on the Menu Bar).

1.32 Gad Grid - Defining the Grid-Definition file

The easiest way to look at the Grid-Definition file is as a normal script file that would be passed to the EXECUTE command. It can contain any command that a normal script file would contain, including IF, ELSE etc..

The file actually consists of a number of scripts, each one to be executed when the appropriate gadget is clicked on. Each script is seperated within the file by a line starting with the '#' character.

(This is called a Seperator-Line)

Following the '#' character is some text, this text will be printed in the gadget box. (Or as much of it as will fit in the box)

When GadGrid is run it first counts the number of lines starting with the '#' character. It then creates a grid with this number of gadgets in it.

Then, starting from the Top-Left of the Gadget Grid and working its way down the window from Left to Right, it assigns each section of script to a gadget and prints the text from the Seperator-Line in box for that gadget.

Any text that occurs at the start of the GadGrid file before the first Seperator-Line will be ignored by GadGrid.

[Example of a basic Grid-Definition file](#)

1.33 Example of a basic Grid-Definition file

This Grid-Definition gives you the options of running 3 programs.

```
+-----+
|#Pro Tracker |
|SYS:Music-Tools/ProTracker |
|#Deluxe Paint |
|CD SYS:Graphic-Tools |
|DPAINT |
|#Amiga Base |
|ASSIGN AmigaBase: SYS:Database-Tools/AmigaBase |
|AmigaBase:AmigaBase |
|ASSIGN AmigaBase: REMOVE |
+-----+
```

The first gadget "Pro Tracker" will simply run the program "ProTracker".

We also specify where to find the command with it's full path first.

The second gadget "Deluxe Paint" will firstly set the current directory to the Graphics-Tools drawer. Then it will run the program "DPaint".

Here, DPaint is stored in the Graphics-Tools drawer, so we need not give the path-name for the command.

The third gadget "Amiga Base" runs the AmigaBase program. AmigaBase requires that the assignment "AmigaBase:" is made to point to the directory where the program is stored before it is run. We then run the program using our new assignment as the path-name. Finally after AmigaBase has finished running we remove the Assignment for tidyness.

1.34 Gad Grid - Introduction

GadGrid is a little utility that will bring up a window containing a grid of gadgets on your Workbench Screen. You can then click on these gadgets to launch your favourite programs.

You probably already have some menu programs, there's a few of them about. Before you drag the GadGrid drawer to your Trashcan just take a look at GadGrid's [Features](#) to see what it can do.

Oh, and if you are wondering how GadGrid got its name, it's short for Gadget Grid. (Not obvious is it ?!)

1.35 Gad Grid - Disclaimer

Although every effort has been made to ensure that GadGrid functions as described in this document, it is provided with no guarantee of it being free from errors, or suitable for every system set up. No liability shall be accepted by the author, or any person involved with the distribution of GadGrid, for any loss or damage of any nature suffered as a result of the use or mis-use of this program.

Your license to use this program is only valid if you accept all liabilities for damage to any hardware or stored information caused by the use of this program.

1.36 Gad Grid - Features

Every single byte of GadGrid was meticulously hand-crafted with your satisfaction at heart. It is only with such great care that we are able to offer you such astounding features as listed below. And remember... GadGrid is not available in the shops...

- * GadGrid gets it's list of Gadget Titles and Commands from any plain ASCII text file. The format of this file is very simple to create.
 - * GadGrid can be run from WorkBench or CLI, with full support for both.
 - * When a gadget is clicked on GadGrid can execute a list of commands as a normal AmigaDOS script. This script doesn't have to be created as a separate file, it can be part of the Grid-Definition file.
 - * GadGrid can "Hide" it's window. A small Title-Bar only window is then opened, simply click on this with both mouse buttons to re-open the Main GadGrid window.
 - * GadGrid does not limit you to a set number of gadgets, you can have as many as you can fit on the screen.
 - * The gadget text can be in any font you desire, and in any size. The height of the Gadgets will be adjusted to fit your font.
 - * By default, the width of the GadGrid window is set to the width of your WorkBench screen (640 is not assumed). But you can specify any width you want.
 - * You can specify how many Columns of gadgets there will be in the grid and GadGrid will automatically adjust the width of the gadgets so they fit in the GadGrid window perfectly.
 - * GadGrid is an itsy bitsy teeny weeny program, under 12k uncrunched. There are many more features in GadGrid too, but I don't want to bore you so I'll let you find them out for yourself by reading the actual documentation.
-

1.37 Gad Grid - Installation

Installing GadGrid is easy. Just follow the step by step instructions below..

STEP 1 : Drag the GadGrid program icon to the Drawer where you want it to be installed.

1.38 Gad Grid - Distribution

This program is Freeware. This means that Copyright to this program remains with the Author at all times. However, as long as it remains unchanged and this document stays with it at all times, you may distribute GadGrid in any way you please.

1.39 Gad Grid - About the Author

If you wish to contact me for any reason then you can write to me at this address...

Mike Archer
29 Holbeck Ave
Marton
Blackpool
FY4 4LS
(England)

I welcome ANY comments on GadGrid, if you have any ideas on how it could be improved then write and tell me. Please enclose a SAE if you want a reply.

Anyone distributing Public Domain software is welcome to write to me for an upgrade to the latest version of GadGrid. Just send me one 3.5" Diskette and a SAE envelope (or Jiffy-Bag). I can only offer this upgrade service to distributors and not to users, remember distributors are there to distribute programs and authors to author programs. :-)

1.40 Gad Grid - Known bugs

A lot of time has been spent testing GadGrid to ensure that it is bug free. However, I could never be 100% sure that it will work on every Amiga.

I do not know of any bugs in GadGrid. If however you do find a bug in GadGrid, please write to me and tell me about it. I will take the following action for EVERY bug report received..

- (i) - Include a mention of the bug in this part of the document.
 - (ii) - Try to fix the bug as soon as I possibly can.
-

1.41 Gad Grid - Trouble Shooting

If you encounter any problems using GadGrid, write to me and I will try and solve them for you. (Please include a SAE).

As well as replying to you with the solution, I will also include it in this section for other users to read too.

PROBLEMS... Click on the ? for the solution.

? I have created a Grid-Definition file and set the default tool to "GadGrid", but when I double click on the icon I get an error saying "Unable to open your tool 'GadGrid'"

1.42 Solution

PROBLEM : I have created a Grid-Definition file and set the default tool to "GadGrid", but when I double click on the icon I get an error saying "Unable to open your tool 'GadGrid'"

SOLUTION : When you installed GadGrid, you did not put it in a location where WorkBench will look for DefaultTools. Your best option is to move GadGrid to one of the following locations...

- (1) - The "C:" directory.
- (2) - The "SYS:UTILITIES" directory.
- (3) - The "SYS:TOOLS" directory.

If you do not wish to move GadGrid to one of the above places then there is another option. Edit your StartUp-Sequence file and add in a command to define the Path where GadGrid can be found. The "PATH" instruction should be placed somewhere before the "LOADWB" command. WorkBench will search for it's DefaultTools in any Path that is defined at the time WorkBench is started by the "LOADWB" command.

1.43 Gad Grid - Version History

Version 1.1 (14/5/97)

- * Updated the routine that determines how much of the text will fit in a gadget box to allow for proportional fonts.
 - * Updated the routine that calculates the window height needed to allow for WorkBench having a TitleBar font larger than 8 pixels.
 - * Made GadGrid execute the command scripts as a background task so that
-

the user can click on another gadget straight away without waiting for the previously selected program to finish running.

* Stored position of the Gadget Grid Window when "Hide" is selected.

Now when the Main Window is Hidden, it will re-open at the position it was at when it was closed rather than the position it was at when GadGrid started.

* Included a similar routine to remember the position of the Title-Bar window that is opened when GadGrid goes into "Hide" mode.

* Ensured that the gadgets always fill the width of the window perfectly, this is done in a similar way as the "JUSTIFY" option in word processors.

Version 1.0 (6/5/97)

1.44 Gad Grid - Future Plans

Here are improvements planned for GadGrid in future versions...

* Offer alternatives for the hide option, these may include.

(a) - Create a Icon on the workbench screen.

(b) - Put an entry into the Tools menu on workbench.

(c) - Go completely invisible until a Hot-Key is pressed.

* Allow the user to define the foreground & background colours of the gadget text, both by default and individually for each gadget within the Grid-Definition file.

* Allow keyboard short-cuts to specific gadgets to be defined.

* Allow an option where GadGrid will either exit or hide itself automatically after one gadget click. The "Exit" option will allow GadGrid to be used as a StartUp-Sequence selector program.

Here are features that WON'T be included in future versions, so please don't write suggesting them...

* An inbuilt Editor for changing the Grid-Definitions.

This would make GadGrid too big and as it stands the Grid-Definition file is easily edited in any text-editor you choose.

* Support for CLI arguments on AmigaDOS 1.x

This would be too much like hard work, to be worth the effort. There are already many other options available for configuring GadGrid.

Read the Section on using GadGrid from CLI for more information on this.

1.45 Gad Grid - Problems with power-packer

I'm not entirely sure what is happening, but I think that there is a bug with Power-Packer. (or Power-Packer on my machine at least!)

I crunched GadGrid & then decrunched it again. I compared this final file with the original & found that 2 bytes had been changed in the file. (This was enough to cause the program to crash)

I know it is not GadGrid causing the problems because it also happens on other programs too.

GadGrid works fine when crunched with other crunchers. (But is there any point when the file is under 12k ?!)