

# **URB v1.0 Documentation**

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

	<i>TITLE :</i> URB v1.0 Documentation	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY		April 18, 2022
		<i>SIGNATURE</i>

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

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>URB v1.0 Documentation</b>	<b>1</b>
1.1	URB v1.0 Documentation	1
1.2	Table Of Contents	2
1.3	Requirements	3
1.4	Description	4
1.5	How To Get Started	4
1.6	Running from a CLI	5
1.7	Additional URB bars	5
1.8	Configuration	5
1.9	Commands	6
1.10	Button Type	7
1.11	Label Text	7
1.12	Color	7
1.13	Background	8
1.14	Button Image	8
1.15	Run Type	8
1.16	Command	9
1.17	Stack	9
1.18	Update	9
1.19	Clear	9
1.20	Save	9
1.21	Use	9
1.22	Test	9
1.23	Cancel	10
1.24	Basic Options	10
1.25	Rows of Buttons	10
1.26	Columns of Buttons	10
1.27	Display Order	10
1.28	Async CLI Command	11
1.29	Arexx run Command	11

---

---

1.30	Execute run Command . . . . .	11
1.31	Workbench run Command . . . . .	11
1.32	Advanced Options . . . . .	12
1.33	Borderless Window . . . . .	12
1.34	No Title Borderless . . . . .	12
1.35	Backdrop Window . . . . .	12
1.36	Use Scrollbars . . . . .	13
1.37	Horizontal Weight . . . . .	13
1.38	Vertical Weight . . . . .	13
1.39	Background Image . . . . .	13
1.40	Menus . . . . .	14
1.41	Technical Notes and Warnings . . . . .	14
1.42	Stacksize . . . . .	14
1.43	Program and Port Names . . . . .	14
1.44	Datatypes . . . . .	15
1.45	MUI Control Codes . . . . .	15
1.46	Hints . . . . .	15
1.47	History . . . . .	15
1.48	Bugs . . . . .	16
1.49	Translators . . . . .	16
1.50	Thanks . . . . .	16

---

# Chapter 1

## URB v1.0 Documentation

### 1.1 URB v1.0 Documentation

URB v1.0 Documentation

Copyright ©1997-98 by Dick Whiting

Standard Disclaimer: I wrote it, it works for me, I don't guarantee that it will do anything productive for anyone else, etc. etc. ;-)

HOWEVER, if you do find a use for it: I homeschool my kids and they would love a postcard from where EVER you live. Instant Geography Lesson;)

Postcards: Dick Whiting  
28590 S. Beaver creek Rd.  
Mulino, Oregon 97042  
USA

Email: Dick Whiting <dwhiting@europa.com>  
Home page: <http://www.europa.com/~dwhiting/>

Table Of Contents

Requirements

Description

How To Get Started

Configuration

Menus

Technical Notes and Warnings

Hints

History

---

Bugs

Translators

Thanks

## 1.2 Table Of Contents

MAIN
URB v1.0 Documentation
1.
Requirements
2.
Description
3.
How To Get Started
3.1.
Running from a CLI
3.2.
Additional URB bars
4.
Configuration
4.1.
Commands
4.1.1.
Button Type
4.1.2.
Label Text
4.1.3.
Color
4.1.4.
Background
4.1.5.
Button Image
4.1.6.
Run Type
4.1.7.
Command
4.1.8.
Stack
4.1.9.
Update
4.1.10.
Clear
4.1.11.
Save
4.1.12.
Use
4.1.13.
Test
4.1.14.
Cancel
4.2.
Basic Options

---

- 4.2.1. Rows of Buttons
- 4.2.2. Columns of Buttons
- 4.2.3. Display Order
- 4.2.4. Async CLI Command
- 4.2.5. Arexx run Command
- 4.2.6. Execute run Command
- 4.2.7. Workbench run Command
- 4.3. Advanced Options
  - 4.3.1. Borderless Window
  - 4.3.2. No Title Borderless
  - 4.3.3. Backdrop Window
  - 4.3.4. Use Scollbars
  - 4.3.5. Horizontal Weight
  - 4.3.6. Vertical Weight
  - 4.3.7. Background Image
- 5. Menus
- 6. Technical Notes and Warnings
  - 6.1. Stacksize
  - 6.2. Program and Port Names
  - 6.3. Datatypes
  - 6.4. MUI Control Codes
- 7. Hints
- 8. History
- 9. Bugs
- 10. Translators
- 11. Thanks

## 1.3 Requirements

---

This program uses MUI ©1992-97 by Stefan Stuntz. If you don't have it, get it from Aminet:

```
mui38usr.lha util/libs
```

I HIGHLY suggest you also get WBRun by Sylvain Rougier & Pierre Carrette from Aminet:

```
util/cli/WBRun_fix.lha
```

and install it for use with the WBRun option in URB. This allows running Workbench programs from a CLI and using its TOOLTYPES.

FILES INCLUDED IN THIS ARCHIVE:

Catalogs	- directory containing localization pieces
ReadMe	- short starter version
URB	- the REAL program.
URB.info	- for running from an icon
URB.guide	- this documentation
URB.guide.info	- multiview guide
UrbGrabs	- directory of screen grabs to give you ideas

There may be versions of these with a suffix such as '.DK' this indicates a Danish version for that file. Over time support of additional languages will be added.

## 1.4 Description

This program allows you to put a button bar for executing ARexx scripts, CLI programs, WorkBench programs, and/or Amiga DOS scripts on any public screen (actually, you could put several on each screen).

The button bars that you can create are only limited by your imagination. You can have image buttons with and without text, colored buttons, buttons in columns, across the top and bottom of the screen, down the sides, or arranged in a frame around another applications window. The combinations are virtually endless.

Shortcut keys, MUI control codes for colors and images are also supported. See the example screen grabs for a few of the possibilities.

There is bubble help available for all gadgets, lists, etc. Place the pointer above the item and wait;)

## 1.5 How To Get Started

Unload all of the URB pieces to a directory of your choice and ↔  
click on  
the URB icon. The first time you start an URB it will complain about



not finding its configuration and preferences files--ignore the messages as you have not yet created them.

The

Configuration

options are pretty self explanatory. Play with them to see how they work and what they do.

From the WorkBench screen you can drop icons on the configuration list. Remember to

Update

the entry and save the configuration file.

The only supported tooltypes and CLI options are:

CONFIG=filename    to set which configuration file to use  
                       default is PROGDIR:programname.cfg

HELP=filename      to set where this guide file is  
                       default is PROGDIR:URB.guide

Running from a CLI

Additional URB bars

## 1.6 Running from a CLI

Run >nil: URB [CONFIG=filename] [HELP=filename]

## 1.7 Additional URB bars

To create a new URB bar, just copy this program and icon to a new name, configure it using MUI and the builtin

Configuration

options,

and you can "instantly" have a toolbar for any screen/application you desire. The buttons support shortcut keys by specifying an underscore ahead of the character in the label you wish to make the shortcut.

Example: `_Yam` for a button label makes "y" the shortcut

Note: if you specify more than one label with the same shortcut key, then all of them will be started. This CAN be used to your advantage.

## 1.8 Configuration

---

You may open the Configuration window either from the menu or by using RAMiga-O.

Commands

Basic Options

Advanced Options

## 1.9 Commands

This is the main piece of URB. Here you specify which commands, ← their type of execution, how the button will be displayed, etc. The list can be rearranged using MUI drag and drop facility.

After installing WBrun from Aminet, the fastest way of configuring most of your buttons will be to drop icons on the list portion of this page. Modify the settings as desired, highlight the entry where you want the command to appear, and press Update.

To edit an existing entry, double-click on it, modify the options, and remember to

Update  
it.

Button Type

Label Text

Color

Background

Button Image

Run Type

Command

Stack

Update

Clear

Save

Use

Test

---

Cancel

## 1.10 Button Type

Use the cycle gadget to choose to have a button with Text on it or just an image.

## 1.11 Label Text

This is the text that will appear on a Text button. If the button is an image button, this text will appear in the command list, but not on the button.

The buttons support shortcut keys by specifying an underscore ahead of the character in the label you wish to make the shortcut.

Example: `_Yam` for a button label makes "y" the shortcut

Note: if you specify more than one label with the same shortcut key, then all of them will be started. This CAN be used to your advantage, such as starting YAM and an URB bar for YAM at the same time.

Note: Shortcut keys do NOT work with Image buttons.

The buttons also support a method of specifying MUI control codes so that you can have colored text and MUI images. The '^' character is translated to ESCAPE. A few examples:

```
^P[15]Text    will make 'Text' appear in pen color 15.  
^bText        will make 'Text' appear as bold  
^I[6:30]Text  will place the MUI right arrow before 'Text'
```

You can specify multiple combinations such as bold, italic, colored. These can also be used in addition to the shortcut key. See the MUI docs for possible options and the correct formats.

## 1.12 Color

Pressing this button opens an MUI color requester. The center of the button shows the background color currently selected in the requester; the outer portion of the button shows the background color of the currently selected command list entry. Open the requester and select a color using the RGB page. The MUI and Colormap options are NOT supported.

---

## 1.13 Background

This cycle gadget is used to select either a default background for the button or to use a colored or image background. If you specify both for a text button, the image is used in preference to the color. You can also use a custom background color with an image button. This results in the chosen color appearing as a border around the image.

To return the background color to the MUI default, it is necessary to edit the entry (double-click on it), set background to default, and press the Update button.

There are MANY combinations possible of button type, background, color, and image. Play with them.

## 1.14 Button Image

Here is where you specify what image to appear on an image button or as a background on a text button. Any datatypes image should work, including gif, jpeg, icon, iff, etc.

The images are not scaled so you may have to modify their size in an image program. Images smaller than the button they appear on are tiled, thus making web background images a good choice.

## 1.15 Run Type

Here you specify what type of command is to be executed.

- Label - no command, use for titles, separators on your button bars
  - ARexx - either an ARexx script to be executed or an 'inline' ARexx command. Example "Address YAM show" entered in the Command string gadget would uniconify YAM's screen.
  - CLI - run a CLI program. Enter exactly what you would in a CLI. You should specify complete pathing information and may enter all of the same arguments as you would in a CLI.
  - Execute - execute an AmigaDOS script. Same format as you would enter in a CLI for the script.
  - WBrun - run a program as a Workbench program using the tooltype information in its icon. You need to install WBrun to be able to use this. See  
Requirements
-

## 1.16 Command

Either an inline ARexx command, a program, or a script to be run. Specify complete path information for best results.

## 1.17 Stack

If the program or script you are running has special stack requirements, you can set it here. For WBrun programs, this value is ignored and the stacksize is obtained from the icon.

## 1.18 Update

Update the item HIGHLIGHTED in the command list with the values from the command page gadgets. You MUST press update for changes to be performed.

## 1.19 Clear

Clear the item HIGHLIGHTED in the command list.

## 1.20 Save

Save the current configuration so that it will be used for future executions of this particular URB program. Configurations generally will be programname.config and programname.config.prefs and saved in the directory the program was started from. The configuration window will close after refreshing the button bar.

## 1.21 Use

Makes the current configuration the one used for the remainder of THIS session. The configuration window will close after refreshing the button bar.

## 1.22 Test

Use this one a lot when you are starting out. This will modify the buttons with your choices, but does not save them anywhere.

The configuration window remains open making it easy to see the results of your changes.

---

## 1.23 Cancel

Undo any changes since the last save or use. The button bar will refresh and the configuration window will close.

## 1.24 Basic Options

These are the options that you **MUST** specify to have a running URB button bar. Here you specify how many buttons, what arrangement they are to appear in, and what commands to use in executing the various types of commands.

Rows of Buttons

Columns of Buttons

Display Order

Async CLI Command

Arexx run Command

Execute run Command

Workbench run Command

## 1.25 Rows of Buttons

How many rows of buttons to have for this URB bar.

## 1.26 Columns of Buttons

How many columns of buttons to have for this URB bar.

## 1.27 Display Order

Specify the order for placing commands on the buttons and how they are to appear on the window.

Across - for a standard x across by y down, this will order the commands across the first row, then across the second...

Down - for a standard x across by y down, this will order the commands down the first column, then down the second column...

Top &

Bottom - will place half the rows across the top of the window and the remainder across the bottom of the window. This option combines with the Advanced Options to control the final appearance. If there are an odd number of rows specified, the extra row appears with the bottom group.

Sides - will place half the columns on the left side of the window with the other half on the right side of the window. This option combines with the Advanced Options to control the final appearance. If there are an odd number of columns specified, the extra one appears on the right side.

Frame - will create a frame of buttons surrounding the central rectangle like a picture frame. This one also combines with the Advanced Options to create the final look.

## 1.28 Async CLI Command

Specify what command to use to run CLI programs. This should be an asynchronous command. The default 'c:run >nil:' should be adequate for most people, but you can use others.

## 1.29 Arexx run Command

Specify the command to run ARexx scripts and commands. You would ONLY have to change this if the rx command is NOT in your SYS:rexxc/ directory. I am unaware of any alternatives to this command.

## 1.30 Execute run Command

Specify the command for executing AmigaDOS scripts. I have NO idea why you would ever have to modify this.

## 1.31 Workbench run Command

Specify the command for running workbench programs from their icons. ←

The only one I'm aware of is WBrun (see

---

Requirements  
) . I default  
its location to the C: directory. If you place it somewhere else,  
you will have to modify this setting.

## 1.32 Advanced Options

These options provide the final touches to your URB bar. You can control borders, scollbars, the size of the inner rectangle, and even allow for displaying an image in it for some button arrangements. Most of these options effect each other. Play with combinations until you get it the way you want.

Borderless Window

No Title Borderless

Backdrop Window

Use Scollbars

Horizontal Weight

Vertical Weight

Background Image

## 1.33 Borderless Window

If the window does NOT have scrollbars, this will remove all borders except for the title. If the window does have scrollbars, this removes the left border and the window gadgets.

## 1.34 No Title Borderless

Removes the title from the window. Combine with no scroll bars and borderless to create a bar of just buttons.

## 1.35 Backdrop Window

Specify this to 'glue' the window to the background. You probably don't want to use this until you have the URB bar just the way you want it. A good use for this is a large frame bar with an application such as YAM appearing in the central rectangle. With



Backdrop specified, the YAM windows will never be hidden behind the URB window.

### 1.36 Use Scrollbars

Specify whether to have scrollbars or not. You should have this on while creating your URB bars. You can turn it off after you have just the look you want, if you do not need them.

Note: Shortcut keys do not work for buttons that are hidden.

### 1.37 Horizontal Weight

This controls the width of the central rectangle for the Frame, Sides, and Top & Bottom options of the Display Order . The default value of 10 results in the width of the rectangle equaling the width of the buttons on the two sides. Increasing the value will widen the rectangle, decreasing it will shrink it.

### 1.38 Vertical Weight

This controls the height of the central rectangle for the Frame, Sides, and Top & Bottom options of the Display Order . The default value of 10 results in the height of the rectangle equaling the height of the buttons above and below it. Increasing the value will make the rectangle taller, decreasing it will shrink it.

### 1.39 Background Image

You can specify a datatype's image to place in the central rectangle for the Frame, Sides, and Top & Bottom options of the Display Order . See Button Image for more information.

## 1.40 Menus

Menus are:

Project:

- About - Tells a bit about the various products involved or used.
- MUI - information about Stefan Stuntz's MUI
- URB - brief comments about URB
- Quit - Quit URB

Settings:

- Open - Open configuration window
- Close - Close configuration window
- Load - Load new configuration
- Restore - Restore to LAST saved configuration
- Save - Save the current configuration
- SaveAs - Save the current configuration to a different name
- MUI settings - open the MUI settings window for this application.

## 1.41 Technical Notes and Warnings

This is a random collection of information and warnings.

Stacksize

Program and Port Names

Datatypes

MUI Control Codes

## 1.42 Stacksize

The stack size that URB is started with is the stack used for all programs run. This value is dependent on how URB is started and shows in the STACK string gadget on the configuration window.

If you have one that needs a larger stack, specify the Stack size when configuring the button. This is ghosted for WBrwn programs.

## 1.43 Program and Port Names

Each URB clone will have its own port. The port name will be the program name converted to all UPPERCASE. Due to this, you will NOT want to use any special characters in the URB clone's name.

---

## 1.44 Datatypes

Datatypes AKxxxxxx default to HAM6/HAM8 for version 40.x picture.datatypes. You need to change this to ORDERED (any method of your choice, and speed/quality) The symptom of having this set incorrectly is that images will appear as black or blank.

Datatype akGIF generates Enforcer hits with MUI. I recommend using one of the others that are available. I am using zgif v39.18 and all seems fine. It is on Aminet as ZGIFDT39.18.lha in directory util/dtype.

If you don't have a GFX board, be very careful as to your chipmem use. It has proved rather easy, at least for me, to get addicted to putting lots of images on lots of buttons:)

Some of the icon.datatypes available result in memory leaks when used with MUI. Try yours and observe its behavior. If you are seeing unreasonable amounts of memory not being returned to the system, try a different one.

## 1.45 MUI Control Codes

See  
Label Text  
for how you can insert MUI images, colors, etc. as part of the text on a button.

## 1.46 Hints

REMEMBER to press  
Update  
after setting a button's options.

Remember that you can use one URB to start other URB's.

You can use the same shortcut key as a way to start multiple applications with one key press. MUI will start them in the order that they appear on the buttons.

Use Backdrop to keep an URB behind another application's window.

To copy an entry: double-click on the entry, single-click a different entry, and press Update.

## 1.47 History

v1.0 First release. Not old enough to have a history yet;)

---

## 1.48 Bugs

As of the date of uploading this program, I have stopped being able to break it. This does not mean it is bug free. If you find any, please email me so that I can correct it for others.

Thanks.

## 1.49 Translators

Danish	Henrik Andreasen	larsa@post7.tele.dk
German	Michael Lünse	m.luense@t-online.de

If you would like to provide a catalog for these, email them to me and I'll post it on my web page and include them in the next update.

## 1.50 Thanks

Thank you's to:

All of the shareware, freeware, etc. programmers for the AMIGA who make it possible for US to still use our favorite machine;)

Special thanks to those who managed to break this program until it finally worked:

Stefan Akerblom, Charles Cooper, and John Pullicino.

Dick Whiting  
March 6, 1998

---