

URB V2.0 Documentation

COLLABORATORS

	<i>TITLE :</i> URB V2.0 Documentation	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY		April 12, 2022
		<i>SIGNATURE</i>

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	URB V2.0 Documentation	1
1.1	URB V2.0 Documentation	1
1.2	Table Of Contents	2
1.3	Comment on Version 2	4
1.4	Requirements	5
1.5	Description	5
1.6	Quick Start	6
1.7	Upgrading V1 to V2	6
1.8	Create a New URB	7
1.9	Start the URB v2	8
1.10	Configuration	8
1.11	Commands	9
1.12	Button Type	9
1.13	Label Text	10
1.14	Color	10
1.15	Background	10
1.16	Button Image	11
1.17	Run Type	11
1.18	Command	12
1.19	Stack	12
1.20	Update	12
1.21	Clear	12
1.22	Save	13
1.23	Reload	13
1.24	Quit	13
1.25	Layout	13
1.26	Area Cycles	14
1.27	Cluster Numerics	14
1.28	Subarea Buttons	14
1.29	Cluster Definitions	14

1.30 Cluster number	15
1.31 Rows	15
1.32 Cols	15
1.33 Order	15
1.34 Image	15
1.35 Image File	15
1.36 Window	16
1.37 Borderless Window	16
1.38 No Title Borderless	16
1.39 Backdrop Window	16
1.40 Use Scrollbars	16
1.41 Paths	17
1.42 Async CLI Command	17
1.43 Arexx run Command	17
1.44 Execute run Command	17
1.45 Workbench run Command	17
1.46 Temp Directory	18
1.47 Image Directory	18
1.48 UrbConfig	18
1.49 Urb.guide	18
1.50 UrbConfig Menus	18
1.51 URB the program	19
1.52 URB Project Menu	19
1.53 URB Settings Menu	20
1.54 Display Menu	20
1.55 Lock buttons	20
1.56 Sizing bars	21
1.57 Show All Clusters	21
1.58 Snapshot Window	21
1.59 Unsnapshot Window	21
1.60 Layout Options	21
1.61 Set to Default	22
1.62 Randomize Layout	22
1.63 Reset to last saved	22
1.64 Window Options	22
1.65 Technical Notes and Warnings	22
1.66 Stacksize	23
1.67 Program and Port Names	23
1.68 Datatypes	23

1.69 MUI Control Codes	23
1.70 Hints	24
1.71 History	24
1.72 Bugs	24
1.73 Translators	24
1.74 Thanks	25

Chapter 1

URB V2.0 Documentation

1.1 URB V2.0 Documentation

URB V2.0 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/>
Mail List: To subscribe, email to urb-list-subscribe@makelist.com

Table Of Contents

Comment on Version 2

Requirements

Description

Quick Start

Configuration

URB the program

Technical Notes and Warnings

Hints

History

Bugs

Translators

Thanks

1.2 Table Of Contents

MAIN
URB V2.0 Documentation
1.
Comment on Version 2
2.
Requirements
3.
Description
4.
Quick Start
4.1.
Upgrading V1 to V2
4.2.
Create a New URB
4.3.
Start the URB v2
5.
Configuration
5.1.
Commands
5.1.1.
Button Type
5.1.2.
Label Text
5.1.3.
Color
5.1.4.
Background
5.1.5.
Button Image
5.1.6.
Run Type
5.1.7.
Command
5.1.8.
Stack
5.1.9.
Update
5.1.10.
Clear
5.1.11.
Save
5.1.12.
Reload

- 5.1.13.
- Quit
- 5.2.
- Layout
 - 5.2.1.
 - Area Cycles
 - 5.2.2.
 - Cluster Numerics
 - 5.2.3.
 - Subarea Buttons
 - 5.2.4.
 - Cluster Definitions
 - 5.2.4.1.
 - Cluster number
 - 5.2.4.2.
 - Rows
 - 5.2.4.3.
 - Cols
 - 5.2.4.4.
 - Order
 - 5.2.4.5.
 - Image
 - 5.2.4.6.
 - Image File
- 5.3.
- Window
 - 5.3.1.
 - Borderless Window
 - 5.3.2.
 - No Title Borderless
 - 5.3.3.
 - Backdrop Window
 - 5.3.4.
 - Use Scrollbars
- 5.4.
- Paths
 - 5.4.1.
 - Async CLI Command
 - 5.4.2.
 - Arexx run Command
 - 5.4.3.
 - Execute run Command
 - 5.4.4.
 - Workbench run Command
 - 5.4.5.
 - Temp Directory
 - 5.4.6.
 - Image Directory
 - 5.4.7.
 - UrbConfig
 - 5.4.8.
- Urb.guide
- 5.5.
- UrbConfig Menus
- 6.
- URB the program
 - 6.1.

- URB Project Menu
 - 6.2.
- URB Settings Menu
 - 6.3.
 - Display Menu
 - 6.3.1.
 - Lock buttons
 - 6.3.2.
 - Sizing bars
 - 6.3.3.
 - Show All Clusters
 - 6.3.4.
 - Snapshot Window
 - 6.3.5.
 - Unsnapshot Window
 - 6.4.
 - Layout Options
 - 6.4.1.
 - Set to Default
 - 6.4.2.
 - Randomize Layout
 - 6.4.3.
 - Reset to last saved
 - 6.5.
- Window Options
 - 7.
- Technical Notes and Warnings
 - 7.1.
- Stacksize
 - 7.2.
- Program and Port Names
 - 7.3.
- Datatypes
 - 7.4.
- MUI Control Codes
 - 8.
- Hints
 - 9.
- History
 - 10.
- Bugs
 - 11.
- Translators
 - 12.
- Thanks

1.3 Comment on Version 2

URB v2 is almost a complete rewrite. The URBs are much more configurable, the configuration program is now a separate program, and there are more options. It should be faster, smaller, fancier, and, unfortunately, not quite as intuitive due to the increased complexity.

There are hours of fun built into it... a hidden gadget, "rubber-bandy thingies" for rearranging the layouts, and a "Randomize" function for the

bored or non-creative.

Enjoy :)

1.4 Requirements

These programs use MUI ©1992-98 by Stefan Stuntz. If you don't have it, get it from Aminet: ←

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

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 their TOOLTYPES.

FILES INCLUDED IN THIS ARCHIVE:

```
Catalogs          - directory containing localization pieces
ReadMe            - comments and requirements for URB v2
ReadMe.mui       - information about Stefan Stuntz' MUI
URB              -
                  Urb the program
                  URB.info          - for running from an icon
UrbConfig        -
                  Configuration
                  program for URBs
UrbConfig.info   - its icon
Urb.guide        - this documentation
Urb.guide.info   - multiview guide
UrbUpgrade       - program for
                  Upgrading V1 to V2
                  UrbUpgrade.info   - its icon

Models           - directory of models for creating different URBs
Models.info      - its icon
UrbGrabs         - directory of screen grabs to give you ideas
                  configuration files are in directory Models
UrbGrabs.info    - its icon
```

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.

1.5 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. They can be as simple as a single button or as complex as an almost total Workbench replacement.

You can have image buttons with and without text, colored buttons, buttons that show and hide other buttons, ones that open file requesters, and you can have areas with datatype pictures scattered as you wish. The physical arrangement of clusters of buttons is virtually unlimited.

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;)

1.6 Quick Start

Unpack the archive to where you expect to LEAVE it. This path is ←
used for
some configuration information.

Upgrading V1 to V2

Create a New URB

Start the URB v2

1.7 Upgrading V1 to V2

Double click on the UrbUpgrade icon.

Verify and update the Paths page information. See
Paths
for what
each is used for.

Go to the Upgrade page.

Select a V1 Urb to upgrade and press the UPGRADE button.
You don't need to specify the configuration file if it has the same name
as the V1 Urb program (e.g. URBXXX and URBXXX.cfg)
The old program will be replaced with the new version.
The old configuration and prefs will be merged into the new format,
and a new URBXXX.cfg will be saved. The old files are deleted.

You may upgrade just the configuration of a version 1 URB by checking the "Update Configuration only" gadget and specifying the configuration to be updated. This is useful if you had multiple configurations defined for your version 1 URB. In this case, upgrade does NOT copy the program and its icon.

Upgrade any others you desire.

Start the updated URB, modify the look as desired, and save the changes using the "Save" menu item.

See

Configuration
and
URB the program
for more information.

1.8 Create a New URB

Start UrbConfig

Update "Paths" information. See
Paths
for what each is used for.

Select "Create New Urb" from the menu.

Enter a simple name (e.g. URBYAM, URBMAIN, etc) for the new URB.
The uppercase version of this name will be the URB's ARexx port name.

Specify the directory in which to place the new URB.

Press the OK button to create, Cancel to quit.

Go to the

Layout
page and define the window for the new URB.

The window can be divided into 3 areas. Each area may have 3 subareas. each subarea may have upto 11 "clusters" of buttons and/or image areas. The top and bottom areas will result in clusters arranged across each subarea, the other area allows for placing clusters vertically within the subareas. Play with the options.

You can also import one of the preset models using the "Import Model" menu option. The screen grabs for these are in the directory "UrbGrabs" with file comments describing them.

Press SAVE. This ensures the correct number of command lines are available in the list on the Command page.

Setup some commands on the Command page; note the NEW "Start" type option. This allows one button to hide/show specified "clusters". You can obtain the numbers to enter from the cluster definition window (the one that appears when you press a subarea "button"). You can NOT hide a cluster

from a button WITHIN that same cluster:)

Menu or Button "Save"

See

Configuration
for additional details about UrbConfig.

1.9 Start the URB v2

Start the new URB clone.

Select "Show/Hide
Sizing bars
" from the "Display" Menu

Drag things around, play with them. If you just want to play, select "Randomize" from the menu. You can do this over and over again to see what some of the possibilities are. (RAMiga-Z shortcut).

Select "Save" when you have something you like.

See

Configuration
and
URB the program
for more information.

1.10 Configuration

Configuration is now handled by a separate program named UrbConfig ↔

It may be started by clicking on the UrbConfig icon or from an URB by selecting menu item "Open Configuration" or by using RAMiga-O.

If the correct configuration file has not automatically loaded, select "Load Configuration" from the menu.

Commands

Layout

Window

Paths

UrbConfig Menus

1.11 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 WBrum 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.

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

Update

.

Button Type

Label Text

Color

Background

Button Image

Run Type

Command

Stack

Update

Clear

Save

Reload

Quit

1.12 Button Type

Use the cycle gadget to choose to have a button with Text on it or

just an image.

1.13 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: A shortcut does NOT work if the button is hidden either due to using a "Start" button or scrolling the button out of view. This can be VERY useful in that you can reuse the same shortcut keys as long as you ensure that only one using that key is visible at a 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.

1.14 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.

1.15 Background

This cycle gadget is used to select either a default background \leftrightarrow 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.

1.16 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.

1.17 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 or AmigaDOS command.
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.
- WBrwn - run a program as a Workbench program using the tooltype information in its icon. You need to install WBrwn to be able to use this. See
Requirements
Start - allows you to have a button set to show/hide one or more clusters. The cluster numbers can be determined from the
Cluster Definitions
window that is opened when you click

on a subarea button on the
Layout
page. Numbers are entered
as "1 2 3" or "1,2,3" (without the quotes).

This would show the first of clusters 1, 2, or 3 that was NOT
showing. If ALL were showing, then ALL of them would be hidden.

Note: you can NOT hide a cluster using a button within the
SAME cluster.

1.18 Command

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

The command may now include %d to insert the name of a directory or
%f for the name of a file. Example: "COPY %f TO %d" to create a CLI
command to copy a file to a different directory. This would open a
requester for you to select the input file and then another for the
destination directory.

This field is also used to specify the cluster numbers for "Start"
buttons. The numbers are entered as "1 2 3" or "1,2,3" without quotes.
See

Run Type
for more on Start buttons.

1.19 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.

1.20 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.

1.21 Clear

Clear the item HIGHLIGHTED in the command list.

1.22 Save

Save the current configuration.

Configurations generally will be programname.config and saved in the directory with the corresponding URB program.

If you have the URB running at the same time, you will want to select "Reload" in the URB to see the changes.

1.23 Reload

Reloads the configuration currently loaded. You'll want to do this if you have its URB running and have saved changes from the URB. There is no automatic communication between URBs and the UrbConfig program.

1.24 Quit

Quit without saving any changes.

1.25 Layout

This is where you define what your URB window will look like. It provides for almost unlimited possibilities and, thus, may require a bit of explanation.

Each URB window may be divided into 3 horizontal strips I call AREAS. The simplest URB might have just one, a more complex URB could have all three.

Each area can also be divided into 3 SUBAREAS. These are also divisions across the width of the area.

Each subarea may have upto 11 "CLUSTERS". A cluster may be a group of buttons (1-1500) or an image.

Clusters in subareas of the TOP and BOTTOM area are arranged horizontally; clusters in subareas in the MIDDLE area are arranged vertically.

Area Cycles

Cluster Numerics

Subarea Buttons

Cluster Definitions

1.26 Area Cycles

These set how many subareas are to occur within the area. You may specify values of 0-3. For a simple button bar, you could set the first one to 1 subarea and the other 2 to 0 subareas.

The main portion of this page consists of buttons corresponding to how the window will be divided. Changing the number of subareas updates the number and arrangement of these buttons. It also changes the number of cluster definition gadgets available to set.

1.27 Cluster Numerics

These little numeric gadgets are used to set how many clusters will be defined within each subarea of an area. There may be 0-3 of them visible depending on the setting of the Area Cycles

1.28 Subarea Buttons

There is one button for each subarea defined. The total of all of them provide a basic visual layout of the URB window. Click on one of these to actually define the clusters for that subarea. This will open the

Cluster Definitions
window.

1.29 Cluster Definitions

This is the where the details of the layout are entered. Each cluster may be either a group of buttons or an image area. An image area that does not have a file specified will create a rectangle with the MUI text background.

There will be as many unghosted rows of gadgets as the number of clusters for this subarea. The title of the window displays which area and subarea is being defined.

Press "OK" to make the definition changes or "Cancel" to not make them.

Cluster number

Rows

Cols

Order

Image

Image File

1.30 Cluster number

The number of the cluster being defined. This is the number to use when specifying a "Start" button under

Run Type

This number changes based on adding or subtracting areas, subareas ↔ number of clusters, etc.

1.31 Rows

Number of rows of buttons for this cluster.

1.32 Cols

Number of columns of buttons for this cluster.

1.33 Order

Across will have the button labels arranged across the cluster left to right and top to bottom in rows.

Down will arrange the button labels top to bottom and THEN left to right in columns.

1.34 Image

If checked the cluster will be an image area. If unchecked the cluster will be a group of buttons.

1.35 Image File

The datatype file to be displayed in the image area. You must have the correct datatype to display the image. If a file is NOT defined, the image area will be a rectangle using the MUI background specified for text areas.

1.36 Window

These options provide the final touches to your URB bar. You can control borders, scrollbars, 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.

These options are also available as menu items for each URB.

Borderless Window

No Title Borderless

Backdrop Window

Use Scrollbars

1.37 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.

1.38 No Title Borderless

Removes the title from the window. Combine with no scroll bars and borderless to create a totally borderless window.

1.39 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 a central rectangle. With Backdrop specified, the YAM windows will never be hidden behind the URB window.

1.40 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.

1.41 Paths

This is where you specify the paths for commands, default ↔ directories to search, and other location information.

Async CLI Command
Arexx run Command
Execute run Command
Workbench run Command
Temp Directory
Image Directory
UrbConfig
Urb.guide

1.42 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.

1.43 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.

1.44 Execute run Command

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

1.45 Workbench run Command

Specify the command for running workbench programs from their ↔ icons.
The only one I'm aware of is WBrum (see Requirements). I default

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

1.46 Temp Directory

Directory to use for temporary files. These are VERY small and are deleted almost immediately. Best choice is probably the default "T:"

1.47 Image Directory

Default directory for images. This is the directory that will appear when a filerequester opens for image selection.

1.48 UrbConfig

Full path to the UrbConfig program. This is stored in the configuration files and is referenced when opening the configuration window from an URB.

1.49 Urb.guide

Full path to Urb.guide. This is stored in the configuration files and is referenced when pressing the HELP key from an URB.

1.50 UrbConfig 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
- Load - load an existing URB configuration file
- Reload - reload the current configuration file (see
 - Reload
 -))
- Save - save the current configuration file (see
 - Save
 -))
- SaveAs - save the current configuration file to a new name
- Quit - Quit UrbConfig

Settings:

- MUI settings - open the MUI settings window for this application.
-

Special:

- Trim - will trim the command list to the correct number of entries matching the number of buttons defined through the Layout page. The LAST entries in the list are the ones removed.
- Create - will create a new URB with the specified name. An URB program, its icon, and the current configuration with the new name are saved to the chosen directory.
- Import - will import an existing configuration as a model. There are several provided in the directory Models. The screen grabs for these are in the directory "UrbGrabs" with file comments describing them.

Note: the Path information is NOT imported.

1.51 URB the program

Each URB is a highly personalized program. Besides the

Configuration

options, the look can be modified directly using options available through the URB menus. If you are just starting, you will want to immediately turn on the sizing bars by selecting "Show/Hide

Sizing bars
" from the
Display Menu

.

URB Project Menu

URB Settings Menu

Display Menu

Layout Options

Window Options

1.52 URB Project Menu

- About - Tells a bit about the various products involved or used.
- MUI - information about Stefan Stuntz's MUI
- URB - brief comments about URB
- Load - load an existing URB configuration file

Reload - reload the current configuration file. Use this if you have made changes from UrbConfig or want to undo your unsaved changes.

Save - save the current configuration file

SaveAs - save the current configuration file to a new name

Iconify - iconify this URB

Quit - Quit

1.53 URB Settings Menu

Open config window - will start the UrbConfig program with the current configuration file. If UrbConfig is ALREADY running, it will be uniconified and brought to the front. In this case you may have to LOAD the correct configuration file.

REMEMBER: There is NO automatic communication between URBs and the UrbConfig program.

MUI settings - open the MUI settings window for this application.
See

Hints
for how MUI settings and URB interact.

1.54 Display Menu

This menu allows options useful while altering the look of your URB. ↔

Lock buttons

Sizing bars

Show All Clusters

Snapshot Window

Unsnapshot Window

1.55 Lock buttons

This option toggles whether the buttons are activated or not. Use this while resizing clusters to keep from accidentally running programs. Select it again to unlock the buttons.

1.56 Sizing bars

This option toggles the displaying of the sizing bars. After turning them on, they may be dragged around using the left mouse button. In this mode the objects are resized in a "rubber band" type style.

Shift-drag acts more like the "usual" sizing actions.

Remember the Amiga key method instead of the mouse. When you want to do some very small movements use L-ALT + L-AMIGA and the cursor keys to drag the bars.

The bars are "sticky." If they make contact, they stick together. To "unstick" them, choose Refresh Window from the "Display" menu.

Select

Lock buttons

from the "Display" menu to keep from accidentally having the buttons DO something while resizing.

As always...play with them.

1.57 Show All Clusters

This option toggles showing or hiding ALL clusters. You can use this to force clusters to be displayed. Mainly useful if display problems occur.

1.58 Snapshot Window

Does an MUI snapshot of the window. VERY useful if you have a borderless window and told MUI not to remember on exit.

1.59 Unsnapshot Window

Does on MUI unsnapshot. Included for completeness.

1.60 Layout Options

The arrangement and sizes of clusters is remembered by tracking a series of weighting numbers for each object in the window. These options allow for changing those values.

Set to Default

Randomize Layout

Reset to last saved

1.61 Set to Default

This resets the weighting values for all objects in the window to equal values. You may wish to do this if some objects refuse to be displayed or you need a quick way of starting over.

1.62 Randomize Layout

This option is mostly for fun. It sets every weighting value to a different number. The results are totally unpredictable and can provide hours of amusement. It's most interesting with a large number of clusters defined.

1.63 Reset to last saved

Will restore the weighting values to those of the current configuration. This is a good "undo" after playing around.

1.64 Window Options

This menu provides the same options as
Window

page within

UrbConfig. Changes within an URB are only remembered if "Save" is selected from the

URB Project Menu

.

1.65 Technical Notes and Warnings

This is a random collection of information and warnings.

Stacksize

Program and Port Names

Datatypes

MUI Control Codes

1.66 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 will usually be 4096 bytes if started from an icon, WBstartup, or WBrun.

If you have one that needs a larger stack, specify the
Stack
size

when configuring the button. This is ghosted for WBrun programs.

1.67 Program and Port Names

Each URB clone will have its own ARexx 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.

1.68 Datatypes

Datatypes AKxxxxx 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.

1.69 MUI Control Codes

See
Label Text

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

1.70 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.

See notes in
Label Text
for additional hints.

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.

Remember that if you make changes in either an URB or UrbConfig that the other program will NOT be aware of them. Reload the configuration file in the OTHER program to keep from overlaying your changes.

How close together buttons appear, how close together clusters can be spaced, and how close to the edge of the window clusters may be placed are controlled by your MUI settings. The most important ones of these are found on the "Windows" and "Group" pages of MUI Settings.

1.71 History

V2.0 Total rewrite of the original URB.

1.72 Bugs

Undoubtedly!!

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.

1.73 Translators

Danish	Henrik Andreasen	henrik.andreasen@get2net.dk
German	Michael Lünse	m.luense@t-online.de
Italian	Francesco Mancuso	mcfrank@mediatel.it

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

1.74 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 Charles Cooper who gleefully watched as his miggy crashed time after time.

Charles is also the person to thank (or flame) for the START button idea:)

To all the people that have sent my kids and me postcards:

We have enjoyed EVERY one of them. I apologize for not having responded by email, but have been spending all my spare time working on my programs. Hope you understand and THANK YOU.

To you all on my 50th birthday,

Dick Whiting
June 23, 1998
