FreeWheel

Alastair M. Robinson

COLLABORATORS

	TITLE : FreeWheel		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Alastair M. Robinson	January 18, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

FreeWheel

1.1   FreeWheel V2.1 - ©1999, 2000 Alastair M. Robinson

```
                Another little tool to fine-tune your mouse
  This time especially for WheelMouse owners.

  Copyright ©1999, 2000 by  Alastair M. Robinson
  Freely distributable for non-commercial purposes.

Do you have a wheelmouse?

Are you disappointed with the amount of software which supports
the wheel?

If so, you'll love FreeWheel.

FreeWheel is a commodity which converts wheelmouse events into scrollbar
movement or cursor key movements.  It can also map several useful functions
to the wheel button, and even a fourth button.

FreeWheel is NOT a serial mouse driver - so you'll still need NewMouse
or equivalent to generate the mouse-wheel events in the first place!


                Legal

                Requirements

                Features

                Bugs/Problems

                Configuring

                Credits

                History
```

1.2   Known bugs and problems...

```
Fixed for V1.1:
  The fake RawKey events generated by FreeWheel had an important field
  missing, which could cause CyberGuard hits with MUI programs.

Fixed for 1.1b
  Horizontal scrolling with a 2-axis mouse works again!

Fixed for 2.0
  FreeWheel will no longer attempt to send messages to its own windows.
  This would previously result in a deadlock, since the GUI code doesn't
  check for messages until after the scrolling code has waited for replies!
  This means you can't test the scrolling functions on FreeWheel's own
  sliders.

Fixed for 2.1
  Bus-fault/enforcer-hit removed from PrefsWin.c - no longer tries to extract
  the signal bit from a closed window!

Resolved for 2.1
  FreeWheel and Alessandro Zummo's MUIWheel have both been updated to cooperate
  with each other.  This means that MUI Windows will be ignore by FreeWheel if
  the new version (1.4) of MUIWheel is running.

Resolved for 2.1
  Double-clicking of the fourth button should now work properly.

Problems yet to be resolved:

While FreeWheel has no difficulty scrolling standard gadtools SCROLLER_KIND
gadgets, it will only work with SLIDER_KIND gadgets if the scrolling speed
is set above half.  This is because SLIDER_KIND gadgets quantize their
current position.  Try it out with the WB3.0 Input preferences program
mouse speed gadget, and you'll see the problem; the gadget jumps a little
way, and springs back!

FreeWheel is unable to nudge scrollgadgets created by the following programs:

  KingCon:
    Its scroll gadget is non-standard, and it doesn't listen for
    RawKey events.

  MUI:
    All MUI gadgets are non-standard, and as such don't respond to IDCMP
    messages.  Programs which use cursor up/down for scrolling will work
    however.  Alessandro Zummo has already written a patch for MUI
    (MUIWheel) allowing the most common classes to use the wheel, and
    FreeWheel will no longer attempt to scroll MUI windows if MUIWheel is
    running.

  BGUI:
  ClassAct:
    Again, WheelMouse can't recognise the scrollbars generated by these
    GUI toolkits.  As with MUI, programs which respond to cursor-key
    movements will work.  If you want to see support for wheelmouse events
```

in these packages, pester the authors; adding wheelmouse support should
be very very easy for someone with access to the sourcecode, and very
very hard without!

Not all programs recognise the shift-clicks generated by FreeWheel in
place of clicks of the third and fourth button. Workbench works fine,
Imagine completely ignores the shift-clicks, and DrawStudio recognises
them but doesn't draw handles for shift-selected objects!
(This only applies to Shift+Click mode; using the buttons to simulate just
the shift key works fine.)

1.3   What you need to run FreeWheel...

FreeWheel's system requirements are pretty light –
you need:

» An Amiga (surprise surprise...)

» Kickstart 3.0 or higher

» A mouse (preferably a PC wheelmouse), though FreeWheel version 2 is
   also useful for three-button mouse and trackball owners.

» A driver for your wheelmouse. (Tested with Alessandro Zummo's
   NewMouse from Aminet, Mroocheck from Elbox, and two home-brew
   mouse-port-adapter-type-things (Serial and PS/2 versions), both
   of which might one day appear on Aminet.)

1.4   The obligatory legal mumbo-jumbo...

THIS SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND. ALL USE IS
AT YOUR OWN RISK, AND THE AUTHOR CANNOT BE HELD RESPONSIBLE FOR DAMAGES OF ANY
KIND, ARISING FROM THE USE OR MISUSE OF THE PROGRAM OR ANY OF ITS ACCOMPANYING
FILES.

While this program is copyrighted, this is done merely to prevent others from
claiming it as their own work. The name FreeWheel is intended to imply its
free status – you may pass the complete package on to whoever you wish, on
a non-profit basis.

Full source code is provided, in the hope that it will be of help and interest
to others. You are encouraged to incorporate any of the source files into
your own projects, as long as they are freeware/shareware, and you credit me.

FreeWheel may be placed on the Cover CDs of Amiga magazines, and any other
Amiga CDs which contain predominantly Shareware, Freeware or PD software.

For any other commercial distribution or use of the source,
mail me at:blackfive@fakenhamweb.co.uk to discuss terms.

1.5   How to configure FreeWheel....

FreeWheel is configured through a simple GadTools GUI:
To show the GUI, you can use the Commodites Exchange, run the program a
second time, or use a hotkey (default ctrl alt f).

The gadgets in the main window are as follows:

Mouse speed X/Y:
  These allow you to adjust the mouse's speed with very fine control, just
  like my previous Commodity, FreeMouse. Setting the horizontal and vertical
  mouse speeds separately can be useful if you're working in a screen mode
  that doesn't have square pixels.

Click to front:
  When this is checked, a double-click in a window's border will bring that
  window to the front.

Click to back:
  When this is checked, a double-click in the border of the frontmost window
  will send it to the back. Remember that a window is not necessarily the
  frontmost window just because it isn't obscured by other windows.

Hot key:
  This string gadget contains a description of the commodities hotkey which
  is used to bring up the GUI.

Set Buttons:
  This will open another window, which will allow to you assign functions to
  the third and fourth mouse buttons, as follows:

  Click:
    The following functions can be assigned to a click of either button:

    » Shift - will simulate the left shift key being held down.

    » Shift + Left Button - Simulates a click with the shift-key held down.
      N.B. Not all program will recognise these shift-clicks.

    » Toggle Left Button - Useful for trackball owners; one click will
      simulate a press of the left button, another will release it.

    » Movement -> scroll - Also for trackball and three-button mouse owners:
      While the button is held down, movement events will be converted to
      scroll events.

    » Move/Scroll Toggle - As Movement -> Scroll, except one click will
      enable the conversion, another will disable it.

    » Cycle Screens - The topmost screen will be sent to the back.

  Click + Roll:

    » Depth Arrange Windows - My favourite feature: If you roll the wheel
      away from you, the window under the pointer will be sent to the back,
      and vice-versa.

» Horiztonal Scroll - In this mode, while the button is held down,
  vertical scroll events will be converted into horizontal scroll events.

Double Click:

» Cycle Screens - The topmost screen will be sent to the back.

» Swap Axis - Horizontal and Vertical scrolling events will be exchanged.
  This allows you to scroll horizontally with a mouse that only has one
  wheel.

Set Scrolling:
  This will open a configuration window for scrolling-related features:

  Scroll window:
    This item determines whether scroll events affect the currently selected
    window, or the window under the pointer.

  Move->Scroll speed:
    This is the rate at which scroll events will be generated by the Movement
    to Scroll modes.

  Ignore MUI:
    If Alessandro Zummo's MUIWheelPatch version 1.4 or higher is running and
    this is checked, FreeWheel will not send events to MUI windows. This
    avoids previous problems with MUI windows receiving both cursor-key
    presses and scrolling events.

  Mirror H/V Scroll:
    Checking these gadgets will reverse the direction of scrolling.

  Prop Gadgets:
      When this is checked, FreeWheel will attempt to nudge any proportional
      gadgets (scrollbars or sliders) it encounters.

    HSpeed/VSpeed:
      These are the amounts by which the gadgets will be nudged. For scrollers,
      the settings here should be thought of as fractions of a page. Scrolling
      sliders will only work if these are set above half.

  KeyPresses:
      When this is checked, windows which don't contain recognisable
      proportional gadgets, and which listen for keypresses, will receive
      cursor-key presses.

    Shift paging:
      When scroll events build up quicker than a program can clear the
      keypresses, if this gadget is checked, FreeWheel will clear the
      backlog and send a shift-cursor key event, which many programs use
      for page-up/down.

    Page after n lines:
      The number of events which will trigger the Shift page feature.

1.6 Features....

So what does FreeWheel actually do?

»   New for V2.1 – The direction of scrolling events can be reversed.

»   New for V2.1 – FreeWheel can now be made to ignore MUI windows if
     MUIWheel 1.4 is running.

»   New for V2.0 – It can generate scrolling events for devices which
     don't have a wheel. Useful for trackball or three-button mouse owners.

»   New for V1.1 – The Horizontal and Vertical mouse speeds can now be set
     individually. Useful if you're working a screen mode which doesn't have
     square pixels.

»   New for V1.1 – Screens can now be cycled with a double-click of the
     middle/wheel button.

»   New for V1.1 – Horizontal scrollers and sliders can now be moved
     with the wheel. To do this, you need to have the pointer over the gadget.

»   It allows programs not specifically written for wheelmice to be scrolled
     by the wheel. If the window has a normal proportional gadget, it will
     be nudged, otherwise cursor-key presses will be sent to the window.

»   It allows several useful functions to be assigned to the wheel-button
     and to a fourth button if present.

»   It provides fractional scaling of mouse movements. The mouse speed can
     be adjusted continuously from 33% to 300%, just like my previous
     Commodity, FreeMouse.

»   It provides my personally favoured Click-To-Front behaviour. Double-click
     anywhere within the border of a window and it will jump to the front.

What doesn't FreeWheel do?

»   It does NOT provide a serial mouse driver. You will still need a driver
     for your wheelmouse. FreeWheel has been tested with Alessandro Zummo's
     NewMouse, and FreeWheel watches for newmouse standard events. It also
     works with Elbox's Mroocheck, and I'm told that it works with Power
     Computing's Punchinello.
     It has also been tested with a homebrew Serial-mouse-to-Amiga-Mouseport
     adapter, and another one for PS/2 wheelmice.

1.7 Credits....

FreeMouse was written over the Christmas break of 1999, and updated during
the Spring of 2000 by Alastair M. Robinson.

It was compiled with VBCC 0.7, which is totally brilliant. The provided
build script should work as long as both VBCC and the Native Developers Kit
includes (which can be obtained on the Amiga Developers CD) are installed.

I can be reached at the following address:

Alastair M. Robinson
 The Old Chapel,
  Syderstone,
   Kings Lynn,
    Norfolk,
     PE31 8SD,
       (England)

EMail:blackfive@fakenhamweb.co.uk
or:robinsonb5@talk21.com

Greetings to:

Alessandro Zummo for writing NewMouse, MUIWheel & IntuiWheel.

Steve Hall for allowing me to wire up his Amiga between his PC
and its mouse, to eavesdrop on the communication between the
two!

Everyone on the Amiga-C mailing list.

Everyone who has mailed me regarding this and previous projects of mine,
Particularly (in chronological order) Olivier Fabre, Matthew O'Neill,
Holger Beer, Simon Goodwin, Darek Dulian (of Elbox) and Ian Deaville,
for requesting features or pointing out bugs.

Special thanks to Stéphane Payet for providing the French translation
of the Guide file.

The Amiga community for surviving the Dark Age of Wintel.

1.8 History...

Version 1.0
  First Public release.

Version 1.1
  The horizontal and vertical mouse speeds are now independent.

  CyberGuard hits which occurred with MUI programs fixed. Thanks to
  Holger Beer for spotting this and pointing me towards a solution.

  Added features requested by Simon Goodwin, namely the ability to scroll
  horizontal sliders with the vertical wheel if the pointer is over them,
  and double-clicking to cycle screens.

Version 1.1b
  The horizontal gadget code broke the ability to scroll horizontally with
  a horizontal wheel - now fixed. Thanks to Elbox for pointing this out.

Version 2.0
  Major increase in flexibility: Both the third and fourth buttons can have

options applied to clicks, click+rolls, and double-click.

New features requested by Ian Deaville: Scrolling events can now be
generated from movement. This opens up the world of scrolling to trackball
and three-button-non-wheeled-mouse owners.

The third and fourth buttons can now be set to toggle the left-mouse-button,
again useful for trackball owners.

The GUI has been split in to three distinct pages.

Bug fix: FreeWheel will no longer try to send messages to its own windows.
Previously, this would cause a deadlock, since FreeWheel waits for the
replies before the GUI has a chance to process the messages!
This does mean that you can't test the scrolling on FreeWheel's own sliders!

FreeWheel will no longer nudge disabled gadgets!

Numerous other fairly benign bugs and typing errors removed from the source.

Version 2.1
Stupid bus-fault/enforcer-hit removed from PrefsWin.c - no longer tries to
extract the signal bit from a closed window!

Can now reverse the direction of scrolling for both Horizontal and
Vertical events.

Can now cooperate with MUIWheel, ignoring MUI windows if MUIWheel is
running.

Double-click of fourth button now works properly.

Tidied up several source files, adding missing includes and function
prototypes.

French guide file added, courtesy of Stéphane Payet.