

Good Work Systems

R₂ M a r k I I I

Omnimouse

with **IDI**^(TM) 1.01

Input Device Intelligence

featuring the **STM32**
32-bit Flash Micro Controller by **STMicroelectronics**

USER GUIDE v1.01



Table of Contents

I. Installation.....	4
Files Installation.....	4
Alias folder.....	4
Filter folder.....	4
Html folder.....	4
Images folder.....	4
Macro folder.....	4
Sounds folder.....	5
File use and Alternative Locations.....	5
Running the Configurator.....	5
II. Overview of the Configurator.....	6
Device Menu.....	6
Save Memory to Disc:	6
Load Memory from Disc:.....	6
Clear Memory:	6
Properties:.....	7
General:.....	7
Sensor:.....	7
Keyboard Monitor:.....	8
Installing the Filter:.....	8
Update Firmware:.....	8
Configuration Menu.....	9
Configuration Settings:.....	9
Alias File:.....	9
Macro File:.....	9
Background Image:.....	9
Session Macro:.....	10
LED Brightness:.....	10
Sensor DPI:.....	10
Customizing the Omnimouse settings for a game or application	11
Make the Configuration settings.....	11
Make the button settings.....	12
The Command Functions Pop Up.....	12
Mouse Functions:.....	12
Keyboard Functions:.....	12
Scroll Functions:.....	12
Using the Shift Command with the Omnimouse.....	13
Setting Shift 1 and 2: an example.....	14
Macros:.....	18
Saving your customized Configuration.....	19
Using Save.....	19
Save the memory image.....	20

Design improvements to Mark III Omnimouse Edition

The Mark III Omnimouse includes several design changes to the R2 model in part based on user feedback:

Separate model support for left and right handed users One size does not fit all and we have had many requests from gamers and mouse users alike for “equal support” for left handers. The Omnimouse is now available in both left and right handed models, which are identically mirrored versions.

Buttons re-located “toward fingers” The 4 top buttons are “rotated toward the fingers” by one button, or 15° for improved button access with dedicated left/right models.

New Grip with Tactile Finger “Target” Molding The new Omnimouse grip features molded “target” locators for the fingertips that improve ease of use with the grip button zones.

Gold plated switch contacts Improved compact gold plated pcb contacts for membrane switches provide the ultimate in button responsiveness and conductivity.

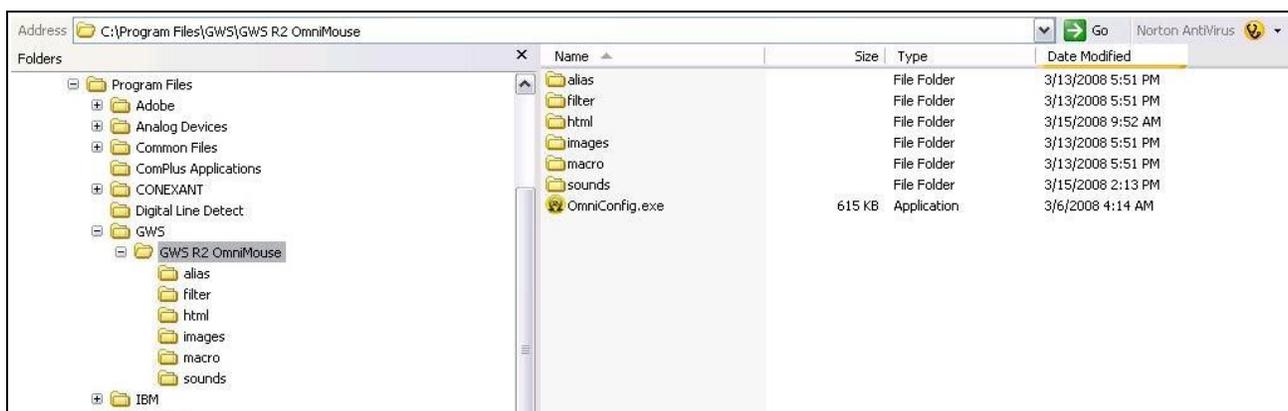
Solid bottom, no detachable base plate or screws Design change to the mouse case makes the Omnimouse even stronger and simpler to maintain. Now only 4 screws.

I. Installation

Download the current version of the Windows Omni Device Configurator from the support page of Gamingmouse.com:

<http://gamingmouse.com/sfiles/OmniDeviceInstall1.0.exe>

Note: This version contains a large sounds file directory (for use with macros). Omnimouse firmware upgrades will be made available in newer versions of the Configurator. When downloading upgrades to the installed Configurator select the version “without sounds” which will update the needed files only.



Files Installation

By default, the installation procedure will install the Omnimouse support files as shown above.

Alias folder

This is where alias files are stored. Alias files are program or game support files that enable the Configurator to display the input commands directly as they are used by the game or application, along with context sensitive help.

Filter folder

IDI technology uses filter files to coordinate other USB devices. These files are located here.

Html folder

Alias and macro files can include html based help that is displayed in the Configurator while editing settings in the device. Html files (containing display help content) linked to macros or alias files are kept here.

Images folder

Images used to skin the Configurator are stored here. Additionally, background images for games or applications to be displayed on a related Configuration page must be placed here.

Macro folder

The macro folder is where all macros available for loading to the Omnimouse are kept.

There are 3 kinds of files in this folder.

IDI Button macros:

(*macroname.idi*) Macros that execute from a button or other user signal. Most macros are this type.

IDI Session macros:

(*sessionname.isi*) Macros that establish settings for a session with a particular game or application.

IDI Macro List files:

(*macrolist.txt*) The files ending with .txt in this directory are special “List” files that allow the Configurator to access and display a set of macros together, usually for a particular game or application.

Macro list files are simple tab-delimited spreadsheet files that can be created and edited with standard spreadsheet programs. They allow you to create your own customized support for each game and keep the macros together.

Sounds folder

IDI technology includes support for embedding sound and voice files in macros useful for creating voice menus, and other feedback. These sounds are kept here.

File use and Alternative Locations

Although these files are part of the software support for the Omnimouse none are required for its basic operations. They support extended capability, and when needed on (another) computer can be run from a USB thumb drive. They do not need to be installed on the hard drive.

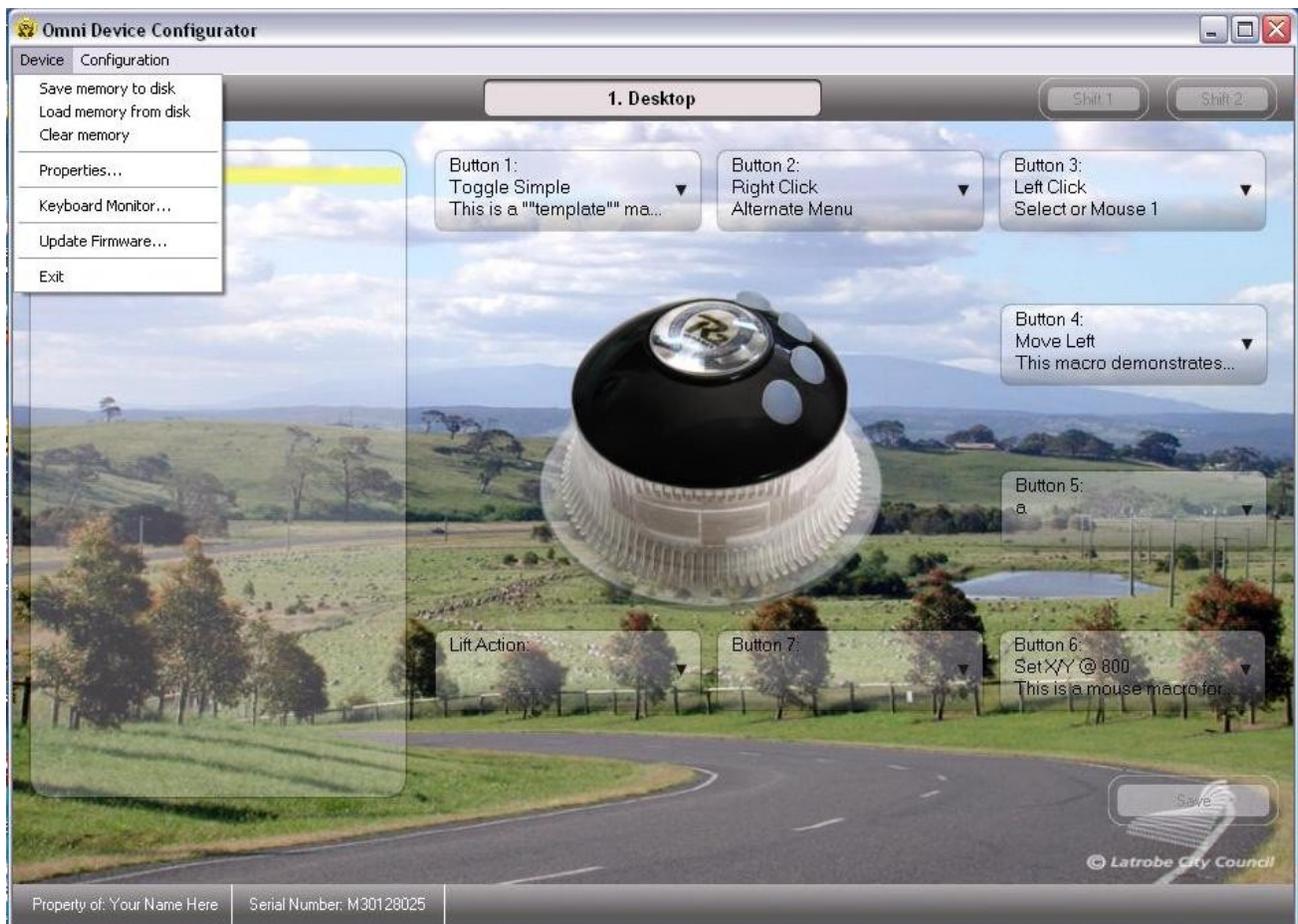
- Macros that are used with the Omnimouse are loaded into device memory, the Macro directory stores the macros when they aren't in memory.
- Sounds aren't stored in the Omnimouse, so access *is* needed to the sounds folder for macros that use sounds
- The Configurator Application remains active only for automatic synchronization with programs (Application Awareness) and the GWS filter driver.

Running the Configurator

The installation process will install an icon on the desktop for activating the Configurator. It will then open the Configurator and minimize it to the system tray. Clicking on either the desktop or tray icon will open the Configurator.

By default the Configurator will load on startup and be available on a click.

II. Overview of the Configurator



The Omni Device Configurator Main Window

The Configurator application is used to make and change Omnimouse settings. It is not required for the normal operation of the Omnimouse which operates independently of PC software. However it does provide special extended functionality used with the Omnimouse's advanced IDI features, and by default it runs as a background application and is minimized in the system tray.

Device Menu

Save Memory to Disc:

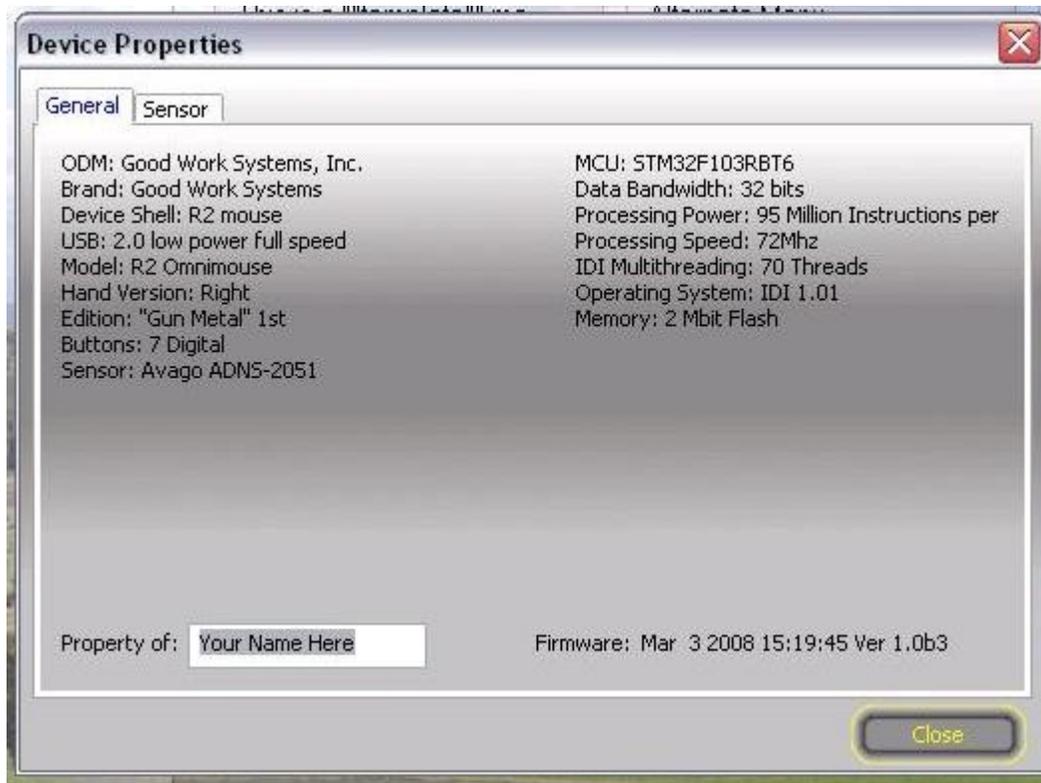
Saves the entire contents of memory to disc. It's recommended that you use this function to back up your configuration information. You can save different "sets" of configurations in device memory, for instance one for applications and one for games, and use the Load Memory command to swap them.

Load Memory from Disc:

Loads a saved memory image back to the device. Allows restoring or swapping configuration information.

Clear Memory:

This clears the memory in the device, and leaves a single left and right click on buttons 1 and 2 in the first profile.



Properties:

General:

Displays the basic hardware and manufacturing specifications for the device.

- **Property of:** Enter your name here, it displays on the main page
- **Firmware:** The current version of firmware is displayed the the lower right corner

Sensor:

The Omnimouse sensor technical information is displayed here.

Keyboard Monitor:



This is the control panel for coordinating a USB keyboard.

Installing the Filter:

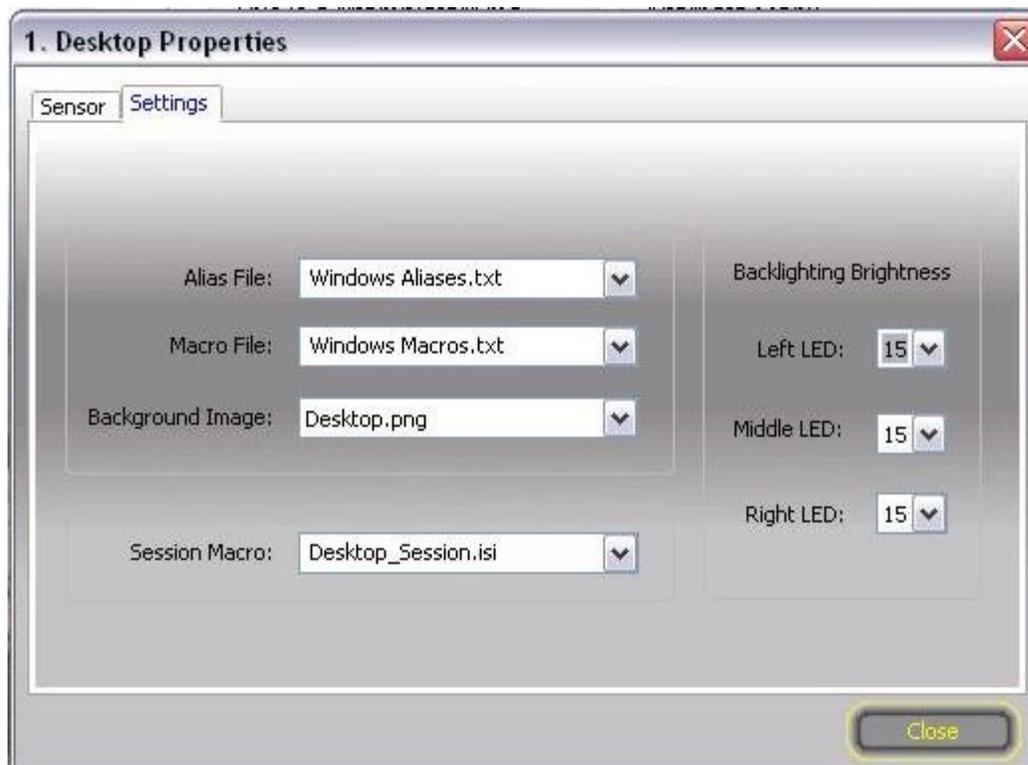
1. Verify that a USB keyboard is plugged to the PC. The keyboard device will be displayed with the word "Not Installed" in red.
2. Click on the install button. The Device will now show as installed. However a message will inform you that the PC must be restarted for the keyboard filter installation to be completed.
3. Close the Configurator and restart the Computer. The keyboard is now synchronized with the Omni Device.

Update Firmware:

Click on this selection to check if there is a new firmware upgrade available with the Configurator.

Configuration Menu

A Configuration is the set of customized settings and programming for the device for use with a particular game or application. Each Configuration has a memory location in the device. The game and application customized settings saved to the device are displayed in the list box on the left of the Configurator. The Omnimouse supports 30 Configurations per memory load. The properties and settings of Configurations are as shown below



Configuration Settings:

Alias File:

Specifies the Alias file to be used with the Configuration. Configurations are game or application specific, so this file would allow the user to bind the device buttons directly to commands of the game or application.

Macro File:

Specifies the Macro file to be used with the Configuration. The macro file specifies a list of macros that can be accessed with this configuration. The macro file could have only specific macros for the game or it could be a larger file with a variety of macros.

Background Image:

Allows specifying an image to display as the “skin” for the Configurator Configuration page. This list displays images in the Image Folder. If you want to use an image for a Configurator background image, be sure to place it in this directory.

The back image should be sized 900 x 540 pixels, and can be jpg, png, bmp or gif image format. If the image is larger (or smaller) it will be scaled for proper display. However, this scaling can result in a slower redraw of the main window.

Session Macro:

Allows specifying a macro that tunes the device settings for a particular game or application.

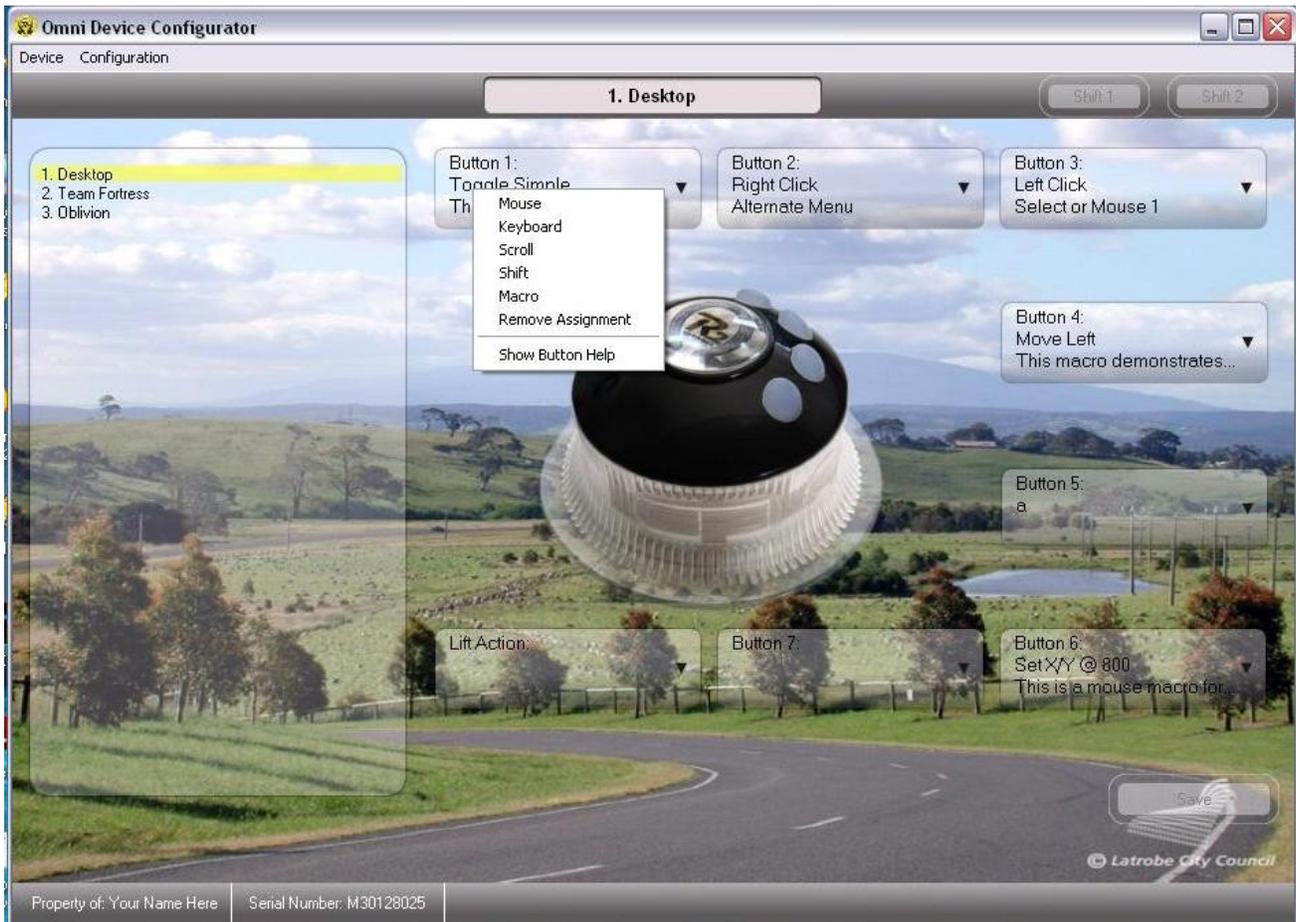
LED Brightness:

Allows setting the brightness of the back lighting when the configuration is active. The illumination sliders can be used to independently vary the 3 LEDs from off to full brightness.

Sensor DPI:

Allows a custom tracking sensitivity setting for the Omnimouse when the configuration is active. This allows you to set the mouse tracking independently for each game or application you use.

Customizing the Omnimouse settings for a game or application



The Configurator with button menu open

Make the Configuration settings

Settings for the Omnimouse are organized by Configuration. A Configuration supports a particular game or application. The list box on the left side of the Configurator displays the game and application settings currently saved in the device. You can use a single Configuration for any and all applications but the Omnimouse allows you to use customized settings with particular applications if you choose.

If you would like to set up the Omnimouse for general use with all applications, you can use the first Configuration factory installed as “1. Desktop”.

If you'd like to use another name, Click on the Configuration Menu and Select Rename.

Name the Configuration

If you want to create a new Configuration to use with a particular game or application, select “New” from the Configuration Menu and enter a name.

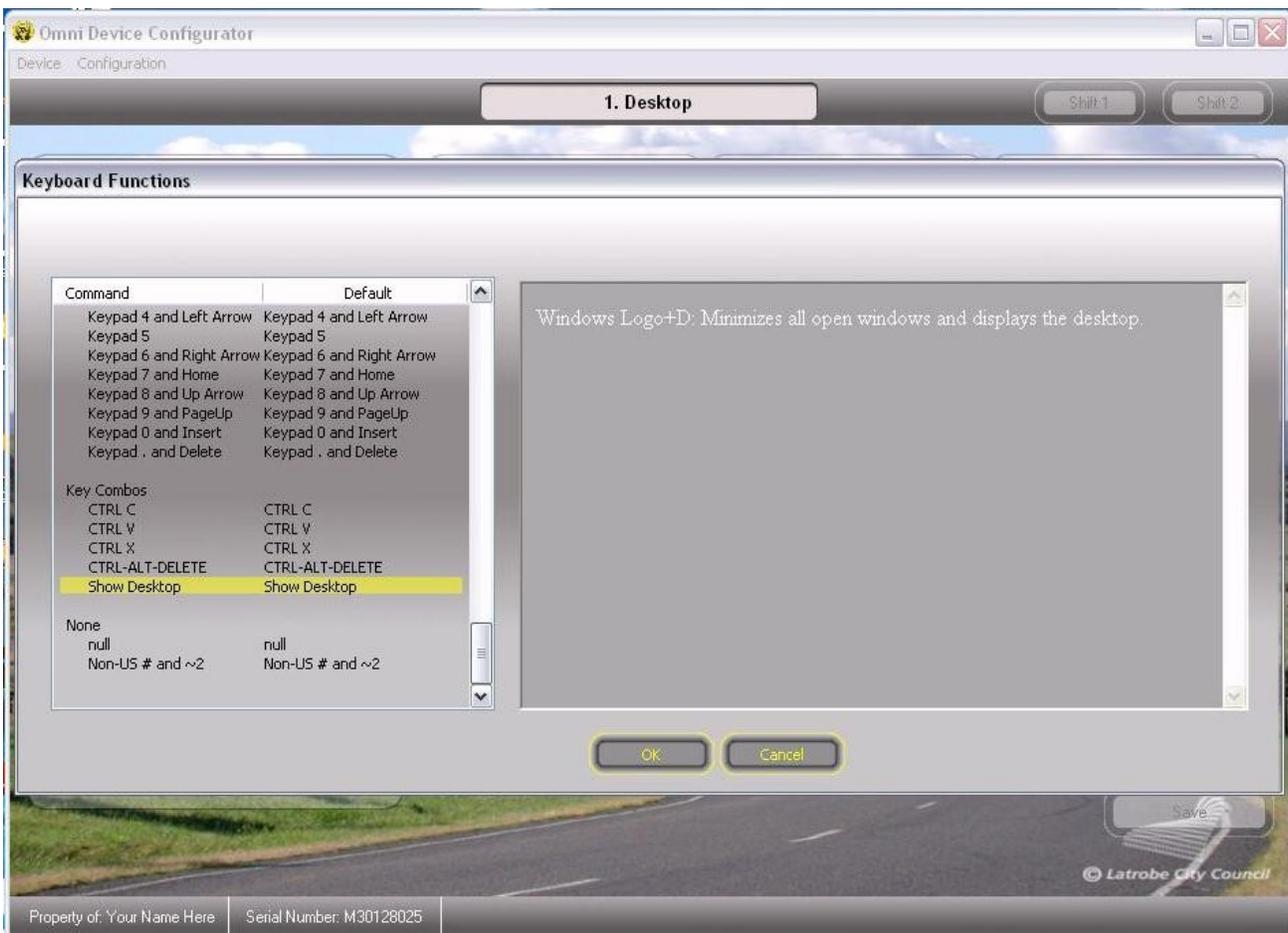
Select the Configuration Settings

Using the Configuration Properties Tabs, make the related settings

- Back lighting LED brightness
- Tracking sensitivity
- Alias File – set Alias file for the game or application if available, otherwise use default
- Macro File - set Macro file for the game or application if available, otherwise use default
- Set Background image for the Configuration page (or use default “gradient.png”)
- Set the session settings for your game or application (DPI, etc)

Make the button settings

Click on the button label for each button and use the button menu to program the button for use specifically with the game or application. There are several categories of commands you can assign.



The Command Functions Pop Up

The Configurator uses a pop-up to display commands for the buttons.

On the left is a scrolling list box with the selections for the command selected.

On the right is a help window which can display help or related information about the selected command.

The help window can show help in 3 different formats.

- Text help (as shown)
- html display help with images and rich text
- website link to site hosted help

Use the Command Functions window to select:

Mouse Functions:

Assign any of the 6 standard Windows button commands to a button.

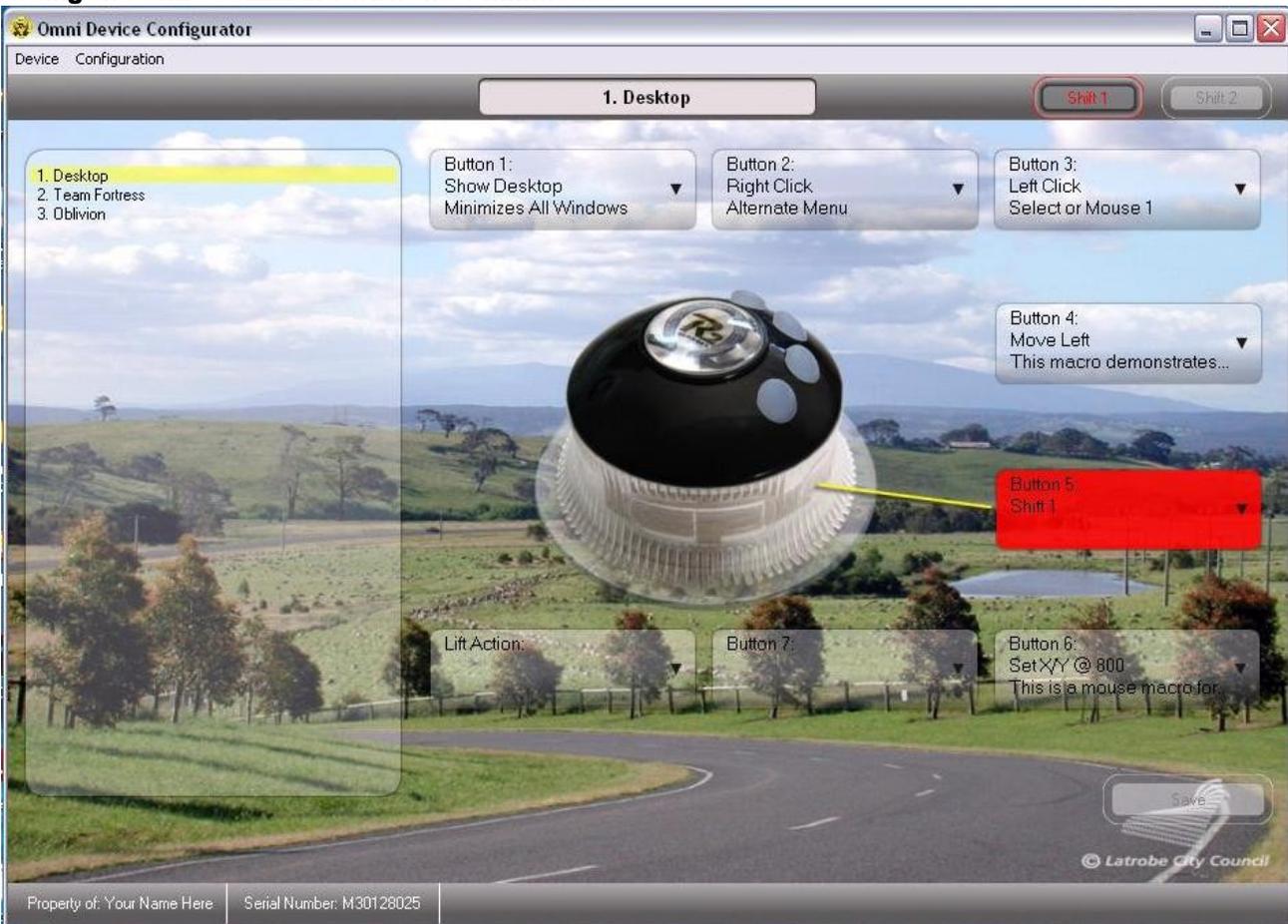
Keyboard Functions:

- Assign any of the 101 keyboard keys including the keypad keys to a button. Pressing the button presses the key.
- Assign any keycombo to a button. Press any or all of the modifier keys (specify right or left set) with any of the keys.

Scroll Functions:

Assign one of a growing number of scroll implementations from simple scrolling buttons to sophisticated (“buttonless” gesture-triggered auto-accelerating scrolling, etc.) Scroll commands are actually IDI macros and can be easily edited and adapted.

Using the Shift Command with the Omnimouse



The Configurator showing button 5 assigned to Shift 1

The Omnimouse Shift function is a powerful button “multiplier”. When a button assigned a Shift function is pressed, the other buttons can send a different, alternate command. Shift functions are designed to be used with the side buttons 5, 6 and 7. These side buttons lie under the grasp and can be squeezed easily and independently of the top buttons.

Shift Icons on Title Bar

There are 2 button Shift commands, Shift 1 and Shift 2. A 3rd Shift is created when both Shift 1 and 2 are pressed together. (Shift 1 + Shift 2 = Shift 3)

On the top right edge of the Configurator Title Bar are 2 Shift icons, which are grayed out when no shift is selected.

- When a Shift button is assigned, the appropriate Shift Icon will appear colored, indicating it is now active.
- The Shift Icons are toggle buttons, that is, they turn the Shift settings on and off.
- Use the Shift Icons to toggle on and off the display of the button settings for each of the Shifts.

Configurator Button Display

Red	Shift 1
Blue	Shift 2
Green	Shift 1 + 2 = Shift 3

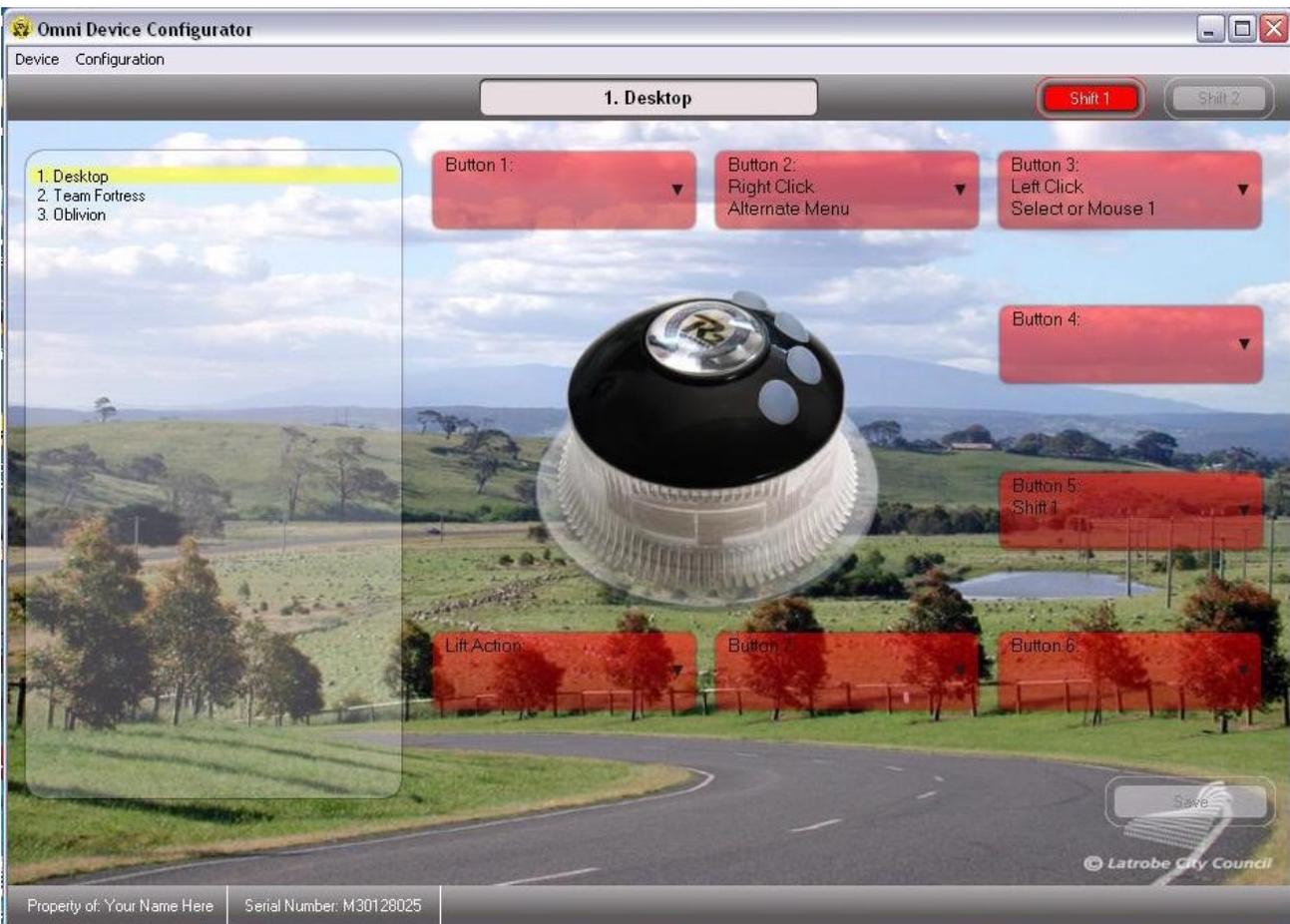
Rules for using the Shift Functions

1. Each Configuration may have (neither), one or both Shifts assigned to buttons.
2. The Shift commands must be assigned to different buttons, they cannot be assigned to the same button.
3. When either Shift button is activated, the other buttons are shifted to their associated Shift commands.
4. When both Shift 1 and 2 buttons are pressed together, a 3rd Shift state is activated, and the Shift 3 button commands become active.
5. Once a button has been assigned a Shift function, it keeps that assignment. It cannot be reassigned other function when another Shift button is pressed.
6. Shift functions can only be assigned to the normal “unshifted” buttons. Shift functions can not be assigned as “shifted” commands.

Setting Shift 1 and 2: an example

The most common assignment for the Shift buttons are:

- Button 5 = Shift 1 (Red) Its easy to squeeze as its “covered” by the middle finger
- Button 6 = Shift 2 (Blue) Its also easy to squeeze with the ring finger when its resting behind the middle finger.
- Squeeze both together for Shift 3.

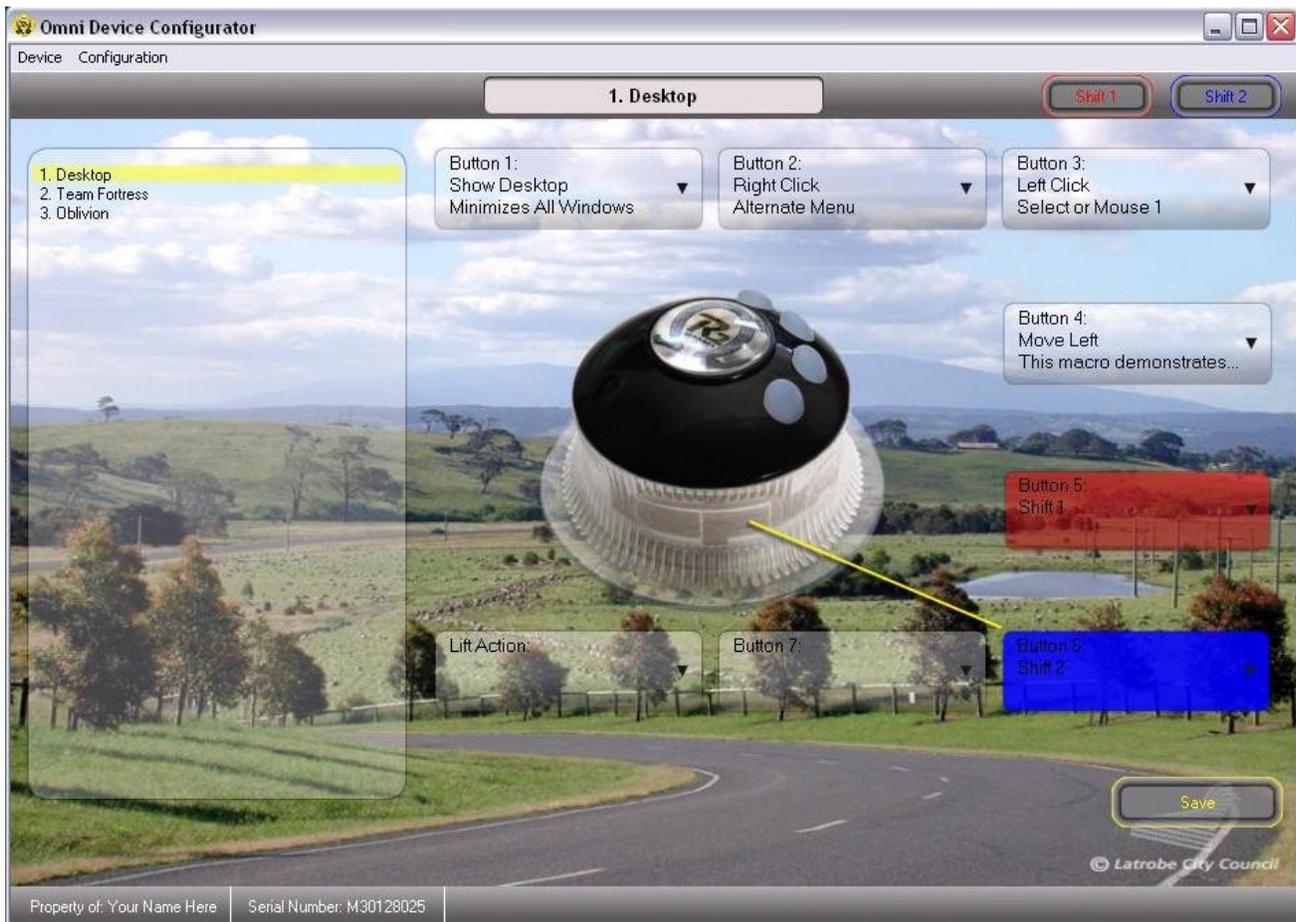


The Configurator showing Shift 1 button assignments with the shift1 icon selected

A. Setting up Shift 1

1. Click on button 5 and select Shift. The first (Shift 1) selection for a Configuration is colored red. Button 5 now appears red.

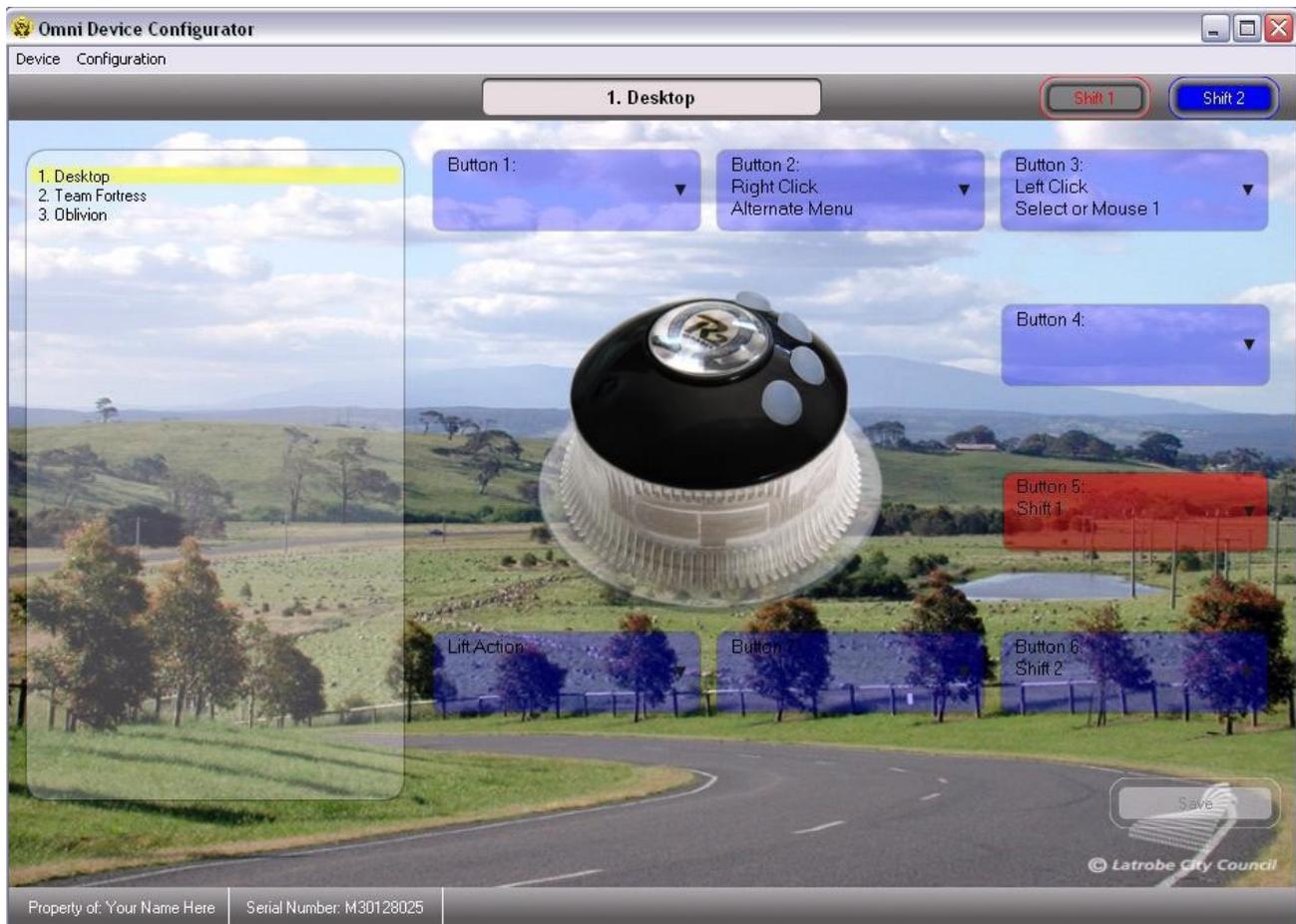
- Click on the Shift 1 indicator on the top title bar which will now be highlighted in red, showing it is active. Note that the button labels turn red.
- Now click on each of these and assign commands. These commands will be active when button 5 is squeezed, and their original commands will be active when button 5 is not squeezed.
- Usually, you will want to keep a few important buttons the same on the shifted and unshifted versions. In effect, this shifts only certain buttons and keeps things fairly simple.



Configurator showing Shift 1 assigned to button 5 and shift 2 assigned to button 6

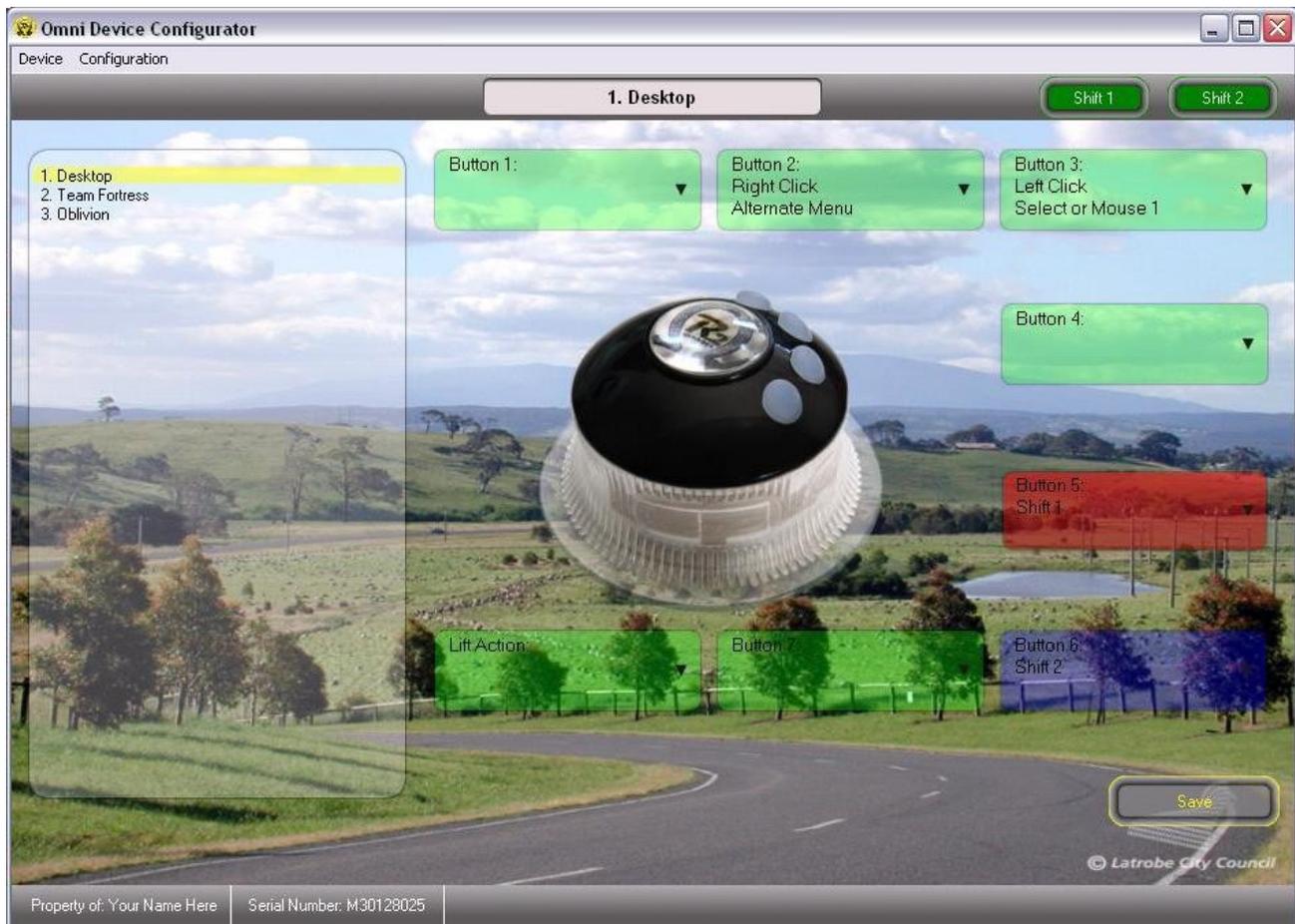
B. Setting up Shift 2

- Click on the Shift 1 red icon to de-select the shift 1 commands display and show the base configuration.
- Click on button 6 and select Shift. The first (Shift 1) selection for a Configuration is colored red. Button 5 now appears red.
- Click on the Shift 1 indicator on the top title bar which will now be highlighted in red, showing it is active. Note that the button labels turn red.



Configurator showing Shift 2 assigned to button 6, and the shift 2 Icon pressed to display Shift 2 settings

4. Now click on each of these and assign commands. These commands will be active when button 6 is squeezed.
5. Again, just like Shift 1, you will want to keep a few important buttons the same on the shifted and unshifted versions. This keeps things fairly simple as you get familiar with IDI Shift functions.



Configurator showing Shift 1 and 2 assigned, and both Shift 1 and 2 icons selected to display Shift 3 settings

C. Setting up Shift 3 (Shift 2 + Shift 3)

1. Click on the Shift 1 red icon and the Shift 2 blue icon to select them **both**. When you do this. They will both turn green, indicating you have selected both and are now displaying the Shift 3 command settings.
2. Now click on each of the buttons and assign commands. These commands will be active when button 5 and button 6 are squeezed together.
3. Once more, you will want to keep a few important buttons the same on the shifted and unshifted versions. You may assign new commands here for only some of the buttons, keeping non shifted commands for others, to keep things simple starting out.

This is the basic procedure for setting the Shift commands.

You can assign macros to buttons in each of the shifted settings to change the Omnimouse back lighting for each Shift state. This gives immediate feedback when using the Shift buttons.

With a little practice, the use of the side buttons to select shifted commands of the other buttons becomes natural and fluid, allowing many different commands to be sent with the same buttons.

Macros:

Macro is another menu choice on the Button Command Menu.

To assign a macro to a button:

- Click on Macro to open the Macro Pop Up window. Its layout is similar to the Command Functions window, and you can scroll the macro list on the left hand side of the window.
- Click on a macro to display its accompanying help text in the help window. This help may be text, html or a web site linked page (viewable if your PC is online).
- Click ok to save the macro to the button, Cancel to close the window without selecting a macro.

What are Macros?

Macros are where your customization ideas become the programming that controls the Omnimouse, the other USB devices and the input session.

Macros take the form of small text files.

Generally, a Macro is a "command unit", it contains the script needed to send the commands the user will send with a particular button press. So macros are usually bound to buttons.

IDI macros have broader power and can control other devices including the keyboard and can be triggered in other ways besides button presses.

Macros = Triggers + Commands

Macros (written in script) contain two kinds of instructions. They contain:

- instructions about *how the macro will be triggered*
- instructions about *what the macro will do when triggered*

Triggers

The instructions in a macro are executed when the macro is **triggered**. The most common trigger for a macro is a button or key press, but IDI enables many other kinds of triggers and combinations of triggers. IDI technology combines high speed mouse tracking analysis with button and key scanning and timer and counting functions to enable a rich set of triggers for macros.

So some of the script in a macro is instructions about **how** the macro is triggered

Commands

IDI technology communicates with the PC by sending input just as a user would. When IDI sends commands, the PC or software application simply sees standard user mouse and keyboard input. IDI can send any kind of keyboard or mouse input including mouse tracking movement, and it can do so while you are also using the mouse and keyboard.

So some of the script in a macro is instructions about **what** input commands to send.

Example Macro

The script below is found in the BT_Toggle_Simple.idi macro. This macro, when assigned to a button will cause it to act like a toggle, sending one command on the odd press and another on the alternate press. The toggle button structure that the script provides could be used to send any commands, in this macro it types some words and makes a click sound.

Blue Text = Trigger instructions; do not change these

Red Text = Commands to send; change these to what ever commands desired

THREAD Profile

VAR Count

Reset:

Count = 0

Continue:

WaitForButton(Button, ButtonMask)

IF Count == 0 THEN

TypeText("odd press",1,10,30)

KeyPressHid(00,40,30)

PlayWav("43649__blubdevis__tik_03.wav", 0, 100)

ELSEIF Count == 1 THEN

TypeText("even press",1,10,30)

KeyPressHid(00,40,30)

PlayWav("43682__stijn__Click6a.wav", 0, 100)

GOTO Reset

FI

Count = Count + 1

GOTO Continue

END

Saving your customized Configuration

Using Save

The Configurator will notice when you have entered a change to a Configuration by highlighting (yellow) the Save button. You must press this before exiting the Configurator for the change to be read to the device memory. You do not need to press it for every setting, just once when you are ready to test the settings. (Each save is a flash memory write, and flash memory has approx 100,000 writes before it may become unreliable. That works out to 32 saves a day for 5 years, so save carefully ;)

What gets saved in the Omnimouse memory

- Configuration associated **filenames** for the various support files (alias, session, macro, BG image)
- Button **names, command or macro script, and help info** are stored for each button and Shift state for each Configuration
 - If help info is text, up to 1250 characters can be stored (A long paragraph)
 - If help info is html or web site based, the URL is saved

What doesn't

- sound files called by macros. The filenames for these are saved directly in macro script*

*These files should be in their proper folders under the Configurator folder for operation. The Configurator folder itself can be located anywhere including an external USB flash drive. Only the sound files are needed for actual macro execution, the others are to support device configuration sessions.

Save the memory image

Once you have set the button and Configuration settings for the Configuration, its a good idea to save the device memory image. This enables restoring and swapping configurations in the device.