# Alias File Guide

## Introduction to alias Files

IDI devices are multi-configurable to work seamlessly with multiple software application and games. Alias files are used to display the standard input commands by their program specific names.

## using alias files

Chose the operating system default or a specific file for a game or program

## alias file format

Standard tab delimited spreadsheet files properly formatted, contain the alias information for a program

## add keycombos to alias files

Add key combos to your button assignment choices by editing the default alias file.

## creating new alias files

Create new alias files for programs using the default alias file as a template

---

## Introduction to Alias Files

### To start, what is an "Alias"?

"Aliases" are the special names given to generic commands by an application.

For instance, CTRL C means pressing the 'c' and the CTRL key at the same time. This is a generic key press.
Most programs use this generic key combo for an operation called "Copy". So "Copy" is an alias of CTRL C.
Its true many programs use the same alias for CTRL C, (Copy) but in general, most programs have their own names and special meanings for mouse commands, keys and keycombos.

| Key | Application | Alias or special meaning |
|---|---|---|
| a | Word | a means the letter a |
| a | Quake III | strafe left |
| a | DrawIt | open snap menu |
| CTRL A | Word | select all |
| CTRL A | Quake III | nothing, unassigned |

### In the above example, what is a "snap menu"?

Not sure. Yes, aliases aren't just names, they are command references with *special meanings* for each program

### So you need a different alias file for each program?

Right, ideally. Alias files also include program related explanations for the aliases. As noted, not all aliases make sense until you learn the program. So the alias files also contain help to learn them. This helps you learn the program.
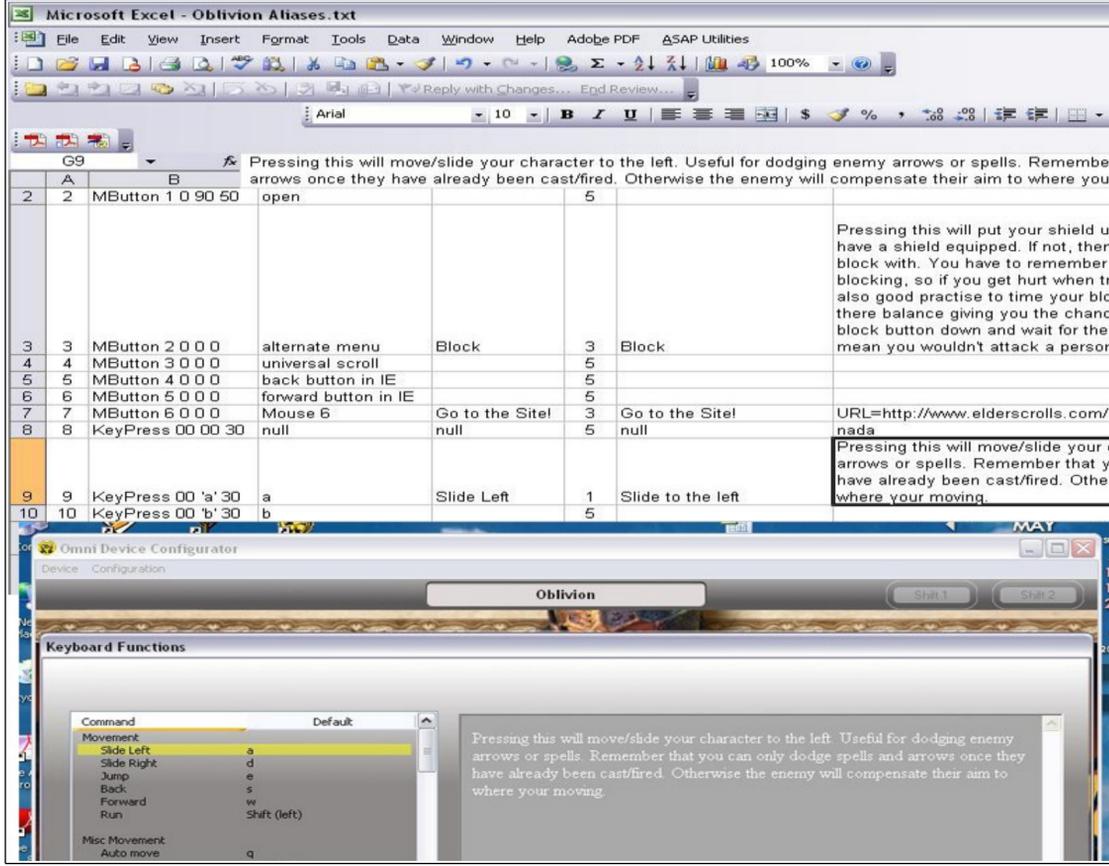


Alias spreadsheet file (*Oblivion*)and related Configurator display

- Alias files are spreadsheet files that connect "behind the scenes" to the Configurator.
- The Configurator is designed so that ordinary spreadsheet files (properly formatted!) supply the command list information that is displayed for the mouse button, scroll action, key and key combo button assignments ( like the Keyboard Functions menu shown.)
- The Configurator supports 30 different device configurations. Each configuration can have its own alias file.
- You can use a different alias file with each configuration, or use the same alias file with multiple configurations.
- As you can see, the alias file is a big table with a row for each of the PC input controls, (all the mouse buttons and keyboard keys) and columns for the program's alias and information for each one.
- Alias files support all mouse button, scroll action, key press and key combo assignments to buttons. More complex assignments are handled by macros.

---

## Alias file use

### What do alias files do exactly?

Alias files control the drop down menus in the Configurator that display mouse, scroll and keyboard commands. Alias files created for specific programs cause the menus to display the program names for the PC input commands instead of the generic ones.

Even the standard keyboard and mouse selections are displayed from a (default) alias file.
Windows Configurator: filename = windows aliases.txt.
Mac Configurator: filename = aliasfile.txt.
These are located in the /alias subdirectory of the OmniDevice installation.
Like any alias files, they can be edited to change or add additional (key combos or scrolling) commands



### What if there isn't an alias file for a program I am using?

Use the default alias file. This will display all the standard inputs with their generic names.
For the Windows Configurator this file is **Windows aliases.txt**
For the Mac Configurator the file is **aliasfile.txt**
You can create an alias file for a program, and GWS will also "host and post" alias files for a variety of programs and games. Check at www.gamingmouse.com for more info.

### How does an alias file get "hooked up" to the Configurator?

Alias files are located in the /alias subdirectory. (You can have as many as you like.)
The files in this directory display in the Alias file Selection Drop Down List on the Configuration Properties page Settings tab for each configuration. This is how you associate one with a configuration.

### Is there a "rule of thumb" for using alias files?

There is no requirement for how you use alias files. They are supplied for your convenience.
Alias files are created to work with a particular program, which means you potentially can have many of them, especially as they are user-creatable.
You can use the default alias file for your configurations, as is. You can optionally edit and "beef up" your default alias file to contain more keycombos and scroll macros, and support your configurations that way.
You can also create or edit an alias file for use with multiple configurations that are used with a similar program, like FPS games, or Video Editors which share keyboard use patterns, (wasd) key combos and other input commands.

### How do I change or reset my alias file for a configuration?

Each configuration has an alias file property, and you set this on the configuration Properties page Settings tab for the configuration.

---

## alias file format

### What is the format for the alias file?

Its a tab delimited spreadsheet file. This means that information in each line is separated by the tab character, and each row ends with a Windows (CR LF) or Mac OS X (LF).

### Does every spreadsheet program support this format?

Pretty much, thats one reason its used. If you don't happen to have a spreadsheet program that does, try the Open Office spreadsheet Calc. Its free and you can download it here.

The spreadsheet at the top of the page shows the alias file contents:

| Index | Code | OS Default | Short Name | Category | Long Name | Help Text |
|---|---|---|---|---|---|---|
| 1 | MButton 1 0 0 0 | Left Click | Attack | 3 | Attack with melee or bow | Pressing this will swing your sword or axe or whatever melee we use a bow, then pressing this will fire off an arrow quickly. Press the attack button will put some 'force' into your attack, with a me as a sword, you will  perform a really hard swing which does a m damage. With a bow, you will pull the arrow right back and it wor until you release the attack button. You must remember to use t |
| 2 | MButton 1 0 90 50 | Double Click | | 5 | | |
| 3 | MButton 2 0 0 0 | Right Click | Block | 3 | Block with shield | Pressing this will put your shield up to block against enemy atta have a shield equipped. If not, then you will put your currently he block with. You have to remember that weapons such as bows a blocking, so if you get hurt when trying to block with a bow, don't also good practice to time your blocks with the enemies strike, o |
| 4 | MButton 3 0 0 0 | Mouse 3 | | 5 | | |
| 5 | MButton 4 0 0 0 | Mouse 4 | | 5 | | |
| 6 | MButton 5 0 0 0 | Mouse 5 | | 5 | | |
| 7 | MButton 6 0 0 0 | Mouse 6 | Go to the Site! | 3 | Go to the Site! | URL=http://www.elderscrolls.com/home/home.php |
| 8 | KeyPress 00 00 30 | null | null | 5 | has no effect | nada |
| 9 | KeyPress 00 'a' 30 | a | Slide Left | 1 | Slide to the left | Pressing this will move/slide your character to the left. Useful for arrows or spells. Remember that you can only dodge spells and |
| 10 | KeyPress 00 'b' 30 | b | | 5 | | |
| 11 | KeyPress 00 'c' 30 | c | Cast | 3 | Cast a spell | Pressing this will cast the currently selected spell. You can use weapon is unsheathed. |
| 12 | KeyPress 00 'd' 30 | d | Slide Right | 1 | Slide to the right | Pressing this will move/slide your character to the right. Useful fo arrows or spells. Remember that you can only dodge spells and have already been cast/fired. Otherwise the enemy will compens where your moving. |
| 13 | KeyPress 00 'e' 30 | e | Jump | 1 | Makes you jump | Pressing this will make you jump. |
| 14 | KeyPress 00 'f' 30 | f | Unsheathe | 3 | Ready weapon | Pressing this will ready your held weapon. Pressing this button w weapon is already unsheathed will put your weapon away. It's ge |

*Alias file for Oblivion in open spreadsheet (top label row is not included, shown for clarity only)*

---

## Alias File General Formating

- Be sure to open and save the file as tab delimited.  This means that information in each line is separated by the tab character, and each row ends with a Windows (CR LF) or Mac OS X (LF).
- Be sure the file ends with .txt and is located in the \alias subdirectory for use with the Configurator
- Clear columns to the right of column G
  Be sure that column G is the last column with data in it.  Be sure column H and above have no contents. Otherwise, columns will not sort correctly in the display
- Be sure not to exceed the character sizes for the columns
- There is a limit of 170 lines to the alias file

## Alias File Format by Column

| | Column | | Max Size | |
|---|---|---|---|---|
| 1 | A | Index | 3 | This is to keep the list in the basic order. If you add rows, you should fill this column in so the rows are properly numbered. |
| 2 | B | Code | 36 | This is the actual code for the control.  It may contain:  MButton() function    (mouse command)  KeyPress() function   (key press or key combo)  filename               (IDI macro ending in .idi)  These 3 options support the alias file selections of a mouse button, scroll action, key press or key combo.  When the alias selection is made, this code gets saved to the button.  There is more info on the code below |
| 3 | C | OSDefault | 16 | This is the name of the control as the operating system refers to it |
| 4 | D | ShortName | 16 | This is the name the program gives to the control.  When a selection is made from the alias file, this becomes the button name, and it gets saved with the code. |
| 5 | E | Category | 2 | This is the category of the program command.  Its used to sort the commands so that they appear in groups related to their function in the program  The Category section at the end of the alias file gives "headers" for these categories.  The commands when displayed in the Configurator will be sorted and displayed under these categories. |
| 6 | F | LongName | 32 | This is a longer name for the command.  This name will be transferred to the button when assigned and displayed on the bottom line of the button label. |
| 7 | G | HelpText | 1,024 | This field is for help text to display for the alias.  Especially for new programs, the aliases may benefit from some explanation. This text  field can contain HTML formatting. and can hold up to 1,024 characters, about 2 fat paragraphs. Or, it can contain special HTML link codes that will allow an html document to be displayed in the help window.  The document can be a local html document in the /html sub directory, or a page hosted by a web site. This also gets saved to the device either as text or a link so the help information continues to remain available.  FILE=filename.html  URL=Http://www.webpage.html |

## Alias File Format by Row

Although it is somewhat arbitrary, there is a standard format for the order of commands by row in the alias file.
- Alias commands (rows) will be sorted in the display by their order in the file.
- The commands will be grouped by Category, and the categories will be sorted by the file order of the Category rows.
- The standard sort order of the alias file is by the Index column.
- The order of the keyboard keys is generally by their underlying HID codes

| Controls | Row Numbers  (Approx) | |
|---|---|---|
| mouse commands | 1-7 | MButton() functions |
| scroll commands | 7-9 | these are macros, and the code column contains the macrofile name. |
| lower case typing keys | 9-61 | KeyPress() functions control these and all the other keyboard commands |
| upper case typing keys | 62-82 | These are standard type characters on the uppercase of the umbers (and others) keys |
| "other" keys | 83-108 | the F1- F10 keys and other non-typing keys |
| modifier keys | 109-116 | The left and right sets of the 4 standard modifier keys, Command, Ctrl, Alt, and Shift |
| keypad keys | 117-133 | |
| key combos | 134-? | This section can be expanded to add commonly used key combos for applications that may not have their own alias file. |
| Categories | last 8 rows | Up to 8 categories supported, these rows contain the name text  for each  category number in the Category column. |

---

## Editing alias files to add keycombos and other assignments

We'll make changes to a copy of the alias file just to be safe.

**STEP 1:** Go to the \alias subdirectory of the Omni device installation directory:
**C:\Program Files\GWS\GWS R2 OmniMouse\alias**

**STEP 2:** Locate the default alias file
For the Windows Configurator this file is **Windows aliases.txt**
For the Mac Configurator the file is **aliasfile.txt**

**STEP 3:** Copy and rename the alias file, leaving it in the directory.  Be sure to use the ".txt" ending

Now we'll open the file with a spreadsheet program as a tab delimited file, and insert some rows

**STEP 1:** Right click on the  (renamed) file name and select "open with".  Then choose a spreadsheet program that supports the tab delimited file format.  Excel and others can be used for this. Open as tab delimited.

**STEP 2:** Widen the columns as needed to display the contents and take a look at the file, using the formating section above for orientation.
Look at column B  the "Code" column. This is the actual code that gets transferred to the button that makes the key press or key combo work.
First thing to notice is that every key that you will use in a key combo is already coded.
To make a new one, you just need to look at the existing KeyPress functions for the proper codes.

**STEP 3:** Insert some rows for the new keycombos.
Go down to the key combo section (starts around row 134, and insert 1 row for each new key combo you want to code.  Of course, you can also edit the existing key combos in place if you only need a few, but it easy to add them, and each alias file supports 24 key combos.
Add these rows before the Category section at the very end of the file.

Now we'll enter a KeyPress() function for the new  key combo in column B for each new key combo

**STEP 1:** The alias file uses a simplified version of the KeyPress function.  It handles both key presses and key combos. There are 4 simple sections separated by a space.

**STEP 2:** Type or paste the word "KeyPress" in the empty column (B) in the row for the key combo.
Be sure to get the capitalized "K" and "P" correctly, the rest can be lower case won't work otherwise.

**STEP 3:** Enter a space and now a 2 number code for the modifier keys that the key combo is going to use. Check the modifier section of the alias for these codes.  They are 2 digit numbers that you add together.

Left CTRL =  01 Left Shift = 02  Left ALT = 04  Left Windows = 08,

the right side versions are just multiplied by 10

Right CTRL = 10 Right Shift = 20  Right ALT = 40  Right Windows = 80

If you want multiple modifier keys for the combo, just add their values.

If you want CTRL + ALT, use 05 (their values added)
If you want CTRL + ALT + Shift, use 07

Just be sure you end up with a 2 digit number.  That's the modifier key part of the code. Of course if you aren't using any modifier keys, just make the number "00"

**STEP 4:** Enter a space and the character code for the key to be pressed with the modifier keys.
All you have to do is look over the alias rows to find the key you want, and copy the code.
For keys that "type" their name, its the key surrounded by the ' character as in
'd' or 'w' or 'T'
(By the way, if you use an upper case letter (T) this will cause the shift key to be added to the key combo, you don't have to specify it in the key modifier section)
If its a key that doesn't type its name like F10 then there will be a 2 digit number code for it.
(F10 = 43)
These codes are all in the alias file already, just scan and find them. They are Hexadecimal numbers (meaning they are based on 16, not 10, ) so they use letters for the numbers between 10 and 16. May look strange but its hex and nothing to worry about.

**STEP 5:** Now enter a space and the last data for the function, which is the code that controls how long the key press or combo will be "pressed."
If you want the key combo or key press to "follow the button" it will be bound to, and use its press and release for the key combo timing, enter "00"
Otherwise, enter the time for the press in milliseconds.  This shouldn't be shorter than 30 (experience shows most apps need at least 30 ms to correctly recognize a key combo under impacted performance conditions)  Key combos don't usually need special  timing, so any value between 30 and 100 should cause the key combo to be performed "normally."
Thats it. verify you have the word "KeyPress" followed by 3 sets of 2 digit numbers, separated by spaces.  This is the code that will be transferred to the button and cause the particular key combo press to occur.

Now we add the rest of the info in the row, the labels that will define and display the ey combo on the button it is assigned to.

**STEP 1:** Use the existing key combo rows as examples, and enter related text content
Column C = this just displays in the drop down, its not saved to the button

**Column D** = this transfers to  and becomes the Button Name, limited to 16 characters
**Column E** = this is the category key for the key combo,  use 8 or whatever number the alias file uses for the key combo section (check the Category rows at the very end of the file)
**Column F** = this is the "long name" it will be saved with the button and displays in the bottom line of the button label.  Limited to 32 characters
**Column G** = this the help text field, see details in the formatting section above

**STEP 2:** Fix the row numbering so the rows are consecutively numbered.  Edit the column A value as needed in the key combos and following Category rows.
Strictly, this isn't necessary as the file will simply be displayed in the current order.  So if you didn't resort, the rows will stay where they are even without a proper index number.  But its best to update the Index numbers so that you can sort the table should you need to and keep the new key combos in their location.

Now we can save the file, open it in a configuration, and load the new key combo

**STEP 1:** Save the file, be sure to save as tab-delimited, and be sure its saved in the \alias subdirectory

**STEP 2:** Click on the configuration you want to use the key combos with, then
Click on the Configuration menu and Select Properties
Click on the Settings Tab..

**STEP 3:** With the Settings Tab open, Click on the Alias file selection List box and select your renamed and just edited alias file. (The selection list shows all the alias files currently in the \alias subdirectory..
The alias file will now be active for the configuration. Close the Settings Tab Window and Click on a button label to bring up the Button Assignment Menu.  Click on Keyboard, and the information in the keyboard rows in the alias file will display.  Scroll down and you should find your new key combos.  Select, and save and test.  You can keep editing the alias file to get your key combos working properly.

**Note:** Once you are familiar with the editing process, you can just edit your existing alias file without any special need to create a backup first, etc.  Its pretty fast to open the file, edit, save and reload the file to get new key combos supported.
However,  If you edit and save changes to an alias file without renaming it, you should close the Configurator to "force" it to reload the new version.  Otherwise the alias file will not be re-read and you will not see the changes you have made.
Close the Configurator by Clicking on the Device menu and Selecting Exit.  Be aware that the Configurator only minimizes when you Click on the close button.

## *creating new alias files*

We'll create a new alias file by copying the default alias file, renaming and editing it.

**STEP 1:** Go to the \alias subdirectory of the Omni device installation directory:
**C:\Program Files\GWS\GWS R2 OmniMouse\alias**

**STEP 2:** Locate the default alias file
   For the Windows Configurator this file is **Windows aliases.txt**
   For the Mac Configurator the file is **aliasfile.txt**

**STEP 3:** Copy and rename the alias file appropriately for use with your particular game or program and ensure it's stays in the \alias folder.  Be sure to use the ".txt" ending

Now we'll open the file with a spreadsheet program like OO Calc or Excel as a tab delimited file for editing.

**STEP 1:** Right click on the  (renamed) file name and select "open with".  Then open using  a spreadsheet program that supports the tab delimited file format. Open as tab delimited.

**STEP 2:** Widen the columns as needed to display the contents and take a look at the file, using the formatting section above for orientation.

In general, you should not have to add rows or change any code, as the default alias file has a row for every possible keyboard and mouse button or key and the proper code for itl.
This means that you should not need to change any of the information in the first 3 colums A-C.
What you do want to do is change how these are displayed by editing the information in columns D-G
One of the nice things about using the default alias file to start with is you dont have to edit any commands that arent used by the game or app. Unused commands will appear as their generic versions.  Just be sure to delete the category key in column E so the unused commands sort properly to the end of the alias file.

**STEP 3:** For each row, update column D
This is the basic game or program name for the input command.
Its limited to 16 characters.
Enter the name for the command, bearing in mind this will also become the button name when the command is assigned to a button.

**STEP 4:** For each row, update column the Category code in column E
This number refers to a command category in the game or program that the command belongs to.
Its used to display the commands in groups that make sense to the program instead of generic alphabetical order.
   **A.** Go to the end of the file to the last 8 rows where the default alias information is categorized.  Just overwrite  the command "headers" (in column D) with the top 8 command categories in your game or program.  If you dont have 8 leave some blank
   **B.** Be sure that the category headers are in "priority order" for your game, as they will be displayed in this order.
   **C.** Now, as you edit each row, be sure to update column E with the category number that the command comes under.  If there is no category number in column E for a row, it will display (by default) at the end of the alias file in a category called "not used"
   **D.** So for any rows with commands not used by the game or application, just delete the number in column E.

**STEP 5:** For each row with a command used by the game or app, update column F
This is the "long name" for the command, a longer version that will display on the bottom line of hte button once the command is assigned.  Its just additional info for the command.
Its limited to 32 characters.
Enter a "longer" name for rows containing commands used by the game or application.

**STEP 6:** For each row with a command used by the game or app, update column G
This is the help text field for the command, and its very helpful to include good application related help info here.  This help text can be text or links to text. It can display html content with images and so on.
See the Alias Formatting section above for the full details about how to include informtion in  this field.

**NOTE:** Be sure there is no text in columns H and above, as this will cause display issues.

Now we can save the file, associate it with a configuration and use it.

**STEP 1:** Save the file, (as tab-delimited,) and be sure its saved in the \alias subdirectory

**STEP 2:** Click on the configuration you want to use with the game or application
Click on the Configuration menu and Select Properties
Click on the Settings Tab..

**STEP 3:** With the Settings Tab open, Click on the Alias file selection List box and scroll through the list of alias files to find your new one.  Select it, close the Settings Tab and Click on the Save button..

Your alias file is now integrated to the Configurator and your configuration, and you should see the special command names and help you entered.
test.  You can keep editing the alias file to get your key combos working properly.

**Note:** Once you are familiar with the editing process, you can just edit this alias file just like any other to add keycombos and so on.

Its pretty fast to open the file, edit, save and reload the file to get changes added like new key combos supported.
However,  If you edit and save changes to an exisiting alias file without renaming it, you should close the Configurator to "force" it to reload the new version.  Otherwise the alias file will not be re-read, the older version will remain,  and you will not see the changes you have made.

Close the Configurator by Clicking on the Device menu and Selecting Exit.  Be aware that Clicking on the Close button at the upper right corner only minimizes  the Configurator to the System Tray.
You need to use the Device/Exit command to fully close the Configurator.