Use vim keys (`h`/`j`/`k`/`l`) to navigate this document. Pressing `s` will fit it to window width (`a` to revert). `K` and `J` zoom in and out. `Super+f` to toggle fullscreen. `q` to quit. (These are general binds set for **zathura**, the pdf reader.) `Mod+F1` will show this document at any time. By `Mod`, I mean the Super Key, usually known as "the Windows Key."

FAQs are at the end of this document. **Naturalness** – Remove the border between mind and matter: everything important should be as few keypresses as possible away from you, and you shouldn't have to think about what you're doing. Immersion. **Economy** – Programs should be simple and light on system resources and highly extensible. Because of this, many are terminal or small ncurses programs that have all the magic inside of them. **Keyboard/vim-centrality** – All terminal programs (and other programs) use vim keys when possible. Your hands never need leave the home row or thereabout. **Decentralization** – This system is a web of small, modifiable and replaceable programs that users can easily customize. Capslock is a useless key in high quality space. It's now remapped. If you press it alone, it will function as escape, making vimcraft much more natural, but you can also hold it down and it will act as another Windows/super/mod key. The menu button (usually between the right Alt and Ctrl) is an alternative Super/Mod button. This is to make one-handing on laptops easier.

If you'd like to change any of these keyboard changes, you need only open and change the `remaps` script. All custom scripts in LARBS are located in `~/.local/bin/`. Actually, this should go without saying, but *everything* here can easily be changed. Below in this document, there is information about where to change programs/components.

Additionally, while this isn't a part of the desktop environment, the default editing mode in the shell is using vi bindings. If you want to learn more of this, run `Mod+F2` and type and select the option for "vi mode in shell". This setting can be changed if you don't like it by deleting or commenting out the contents of `~/.config/shell/inputrc`.

To the left, you'll see the numbers of your current workspace/tag(s). On the right, you'll see various system status notifiers, the date, volume, even music and local weather if possible, etc. Each module on the right of the statusbar is a script located in `~/.local/bin/statusbar/`. You can see what they do and modify them from there. I'm sure you can figure it out. You can also right click on the module to see what it does.

The program dwmblocks is what is run to generate the statusbar from those scripts. You can edit its config/source code in `~/.local/src/dwmblocks/` to tell it what scripts/commands you want it to display.

Press `mod+F2` at any time to get a menu of programs to watch videos about streaming directly from YouTube. You can also check the config files for programs which detail a lot of the specific bindings.

The window manager dwm abstractly orders all of your windows into a stack from most important to least based on when you last manipulated it. dwm is an easy to use window manager, but you should understand that it makes use of that stack layout. If you're not familiar, I recommend you press `Mod+F2` and select the "dwm" option to watch my brief tutorial (note that the bindings I discuss in the video are the default dwm binds, which are different (inferior) to those here).

Notice also the case sensitivity of the shortcutsTo type capital letters, hold down the `Shift` key— that might sound like an obvious and condescending thing to tell you, but there have literally been multiple people (Boomers) who have emailed me asking how to type a capital letter since caps lock isn't enabled.
, Be sure you play around with these. Be flexible with the basic commands and the system will grow on you quick. `Mod+Enter` – Spawn terminal (the default terminal is `st`; run `man st` for more.) `Mod+q` – Close window `Mod+d` – dmenu (For running commands or programs without shortcuts) `Mod+j/k` – Cycle thru windows by their stack order `Mod+Space` – Make selected window the master (or switch master with 2nd) `Mod+h/l` – Change width of master window `Mod+z/x` – Increase/decrease gaps (may also hold `Mod` and scroll mouse) `Mod+a` – Toggle gaps `Mod+A` – Gaps return to default values (may also hold `Mod` and middle click) `Mod+Shift+Space` – Make a window float (move and resize with `Mod+left/right click`). `Mod+s` – Make/unmake a window "sticky" (follows you from tag to tag) `Mod+b` – Toggle statusbar (may also middle click on desktop) `Mod+v` – Jump to master window `Mod+t` – Tiling mode (active by default) `Mod+T` – Bottom stack mode (just like tiling, but master is on top) `Mod+f` – Fullscreen mode `Mod+F` – Floating (AKA normie) mode `Mod+y` – Fibbonacci spiral mode `Mod+Y` – Dwindle mode (similar to

Fibonacci) `Mod+u` – Master on left, other windows in monocle mode `Mod+U` – Monocle mode (all windows fullscreen and cycle through) `Mod+i` – Center the master window `Mod+I` – Center and float the master window `Mod+o/O` – Increase/decrease the number of master windows `Mod+r` – lf (file browser/manager) `Mod+R` – htop (task manager, system monitor that R*dditors use to look cool) `Mod+e` – neomutt (email) – Must be first configured by running `mw add`. `Mod+E` – abook (contacts, addressbook, emails) `Mod+m` – ncmpcpp (music player) `Mod+w` – Web browser (Brave by default) `Mod+W` – nmtui (for connecting to wireless internet) `Mod+n` – vimwiki (for notes) `Mod+N` – newsboat (RSS feed reader) `Mod+F4` – pulsemixer (audio system control) `Mod+Shift+Enter` – Show/hide dropdown terminal `Mod+'` – Show/hide dropdown calculator `Mod+D` – passmenu (password manager)

There are nine tags, active tags are highlighted in the top left. `Mod+(Number)` – Go to that number tag `Mod+Shift+(Number)` – Send window to that tag `Mod+Tab` – Go to previous tag (may also use \ for Tab) `Mod+g` – Go to left tag (hold shift to send window there) `Mod+;` – Go to right tag (hold shift to send window there) `Mod+Left/Right` – Go to another display `Mod+Shift+Left/+Right` – Move window to another display `Mod+BackSpace` –Choose to lock screen, logout, shutdown, reboot, etc. `Mod+F1` – Show this document `Mod+F2` – Watch tutorial videos on a subject `Mod+F3` – Select screen/display to use `Mod+F4` – pulsemixer (audio control) `Mod+F6` – Transmission torrent client (not installed by default) `Mod+F7` – Toggle on/off transmission client via dmenu `Mod+F8` – Check mail, if mutt-wizard is configured. (Run `mw add` to set up.) `Mod+F9` – Mount a USB drive/hard drive or Android `Mod+F10` – Unmount a non-essential drive or Android `Mod+F11` – View webcam `Mod+F12` – Rerun kayboard mapping scripts if new keyboard is attached `Mod+`` – Select an emoji to copy to clipboard `Mod+Insert` – Show contents of clipboard/primary selection

I use ncmpcpp as a music player, which is a front end for mpd. `Mod+m` – ncmpcpp, the music player `Mod+.` – Next track `Mod+,` – Previous track `Mod+<` – Restart track `Mod+>` – Toggle playlist looping `Mod+p` – Toggle pause `Mod+p` – Force pause music player daemon and all mpv videos `Mod+M` – Mute all audio `Mod+-` – Decrease volume (holding shift increases amount) `Mod++` – Increase volume (holding shift increases amount) `Mod+[` – Back 10 seconds (holding shift moves by one minute) `Mod+]` – Forward 10 seconds (holding shift moves by one minute) `Mod+F4` – pulsemixer (general audio/volume sink/source control)

I use maim and ffmpeg to make different recordings of the desktop and audio. All of these recording shortcuts will output into `~`, and will not overwrite previous recordings as their names are based on their exact times. `PrintScreen` – Take a screenshot `Shift+PrintScreen` – Select area to screenshot `Mod+PrintScreen` – Opens dmenu menu to select kind of audio/video recording `Mod+Delete` – Kills any recording started in the above way. `Mod+Shift+c` – Toggles a webcam in the bottom right for screencasting. `Mod+ScrollLock` – Toggle screenkey (if installed) to show keypresses

I've mapped those extra buttons that some keyboards have (play and pause buttons, screen brightness, email, web browsing buttons, etc.) to what you would expect.

Dotfiles/settings files are located in `~/.config/`.

Suckless programs, dwm (the window manager), st (the terminal) and dmenu among others do not have traditional config files, but have their source code location in `~/.local/src/` (press `rr` to jump to that directory). There you can modify their `config.h` files or other source code, then `sudo make install` to reinstall.

vim is set to automatically recompile and install these programs whenever you save changes to any `config.h` file (compilation will be nearly instantaneous). You'll have to restart the program to see its effects obviously.

As mentioned above, LARBS makes some keyboard changes with the `remaps` script. These settings may override your preferred settings, so you should open this file and comment out troublesome lines if you have issues.

On fresh install, the Linux audio system often mutes outputs. You may also need to set your preferred default output sink which you can do by the command line, or by selecting one with `pulsemixer` (`Mod+F4`).

Copying and pasting is always program-specific on any system. In most graphical programs, copy and paste will be the same as they are on Windows: `ctrl-c` and `ctrl-v`. In the Linux terminal, those binds have other more important purposes, so you can run `man st` to see how to copy and paste in my terminal build.

Additionally, I've set vim to use the clipboard as the default buffer, which means when you yank or delete something in vim, it will be in your system clipboard as well, so you can `ctrl-v` it into your browser instance, etc. You can also paste material copied from other programs into vim with the typical vim bindings.

The system will always read the file `~/.config/wall.png` as the wallpaper. The script `setbg`, if run on an image will set it as the persistent background. When using the file manager, you can simply hover over an image name and type `b` and this will run `setbg`.

LARBS no longer deploys Xresources by default, but check `~/.config/x11/xresources` for a list of color schemes you can activate or add your own. When you save the file, vim will automatically update the colorscheme. If you'd like these colors activated by default on login, there is a line in `~/.config/x11/xprofile` you can uncomment to allow that.

Or, if you want to use `wal` to automatically generate colorschemes from your wallpapers, just install it and `setbg` will automatically detect and run it on startup and wallpaper change.

LARBS comes with mutt-wizard, which gives the ability to receive and send all your email and keep an offline store of it all in your terminal, without the need for browser. You can add email accounts by running `mw -a your@email.com`. See `man mw` for all the information you need about mutt-wizard.

Once you have successfully added your email address(es), you can open your mail with `neomutt` which is also bound to `Mod+e`. You can sync your mail by pressing `Mod+F8` and you can set a cronjob to sync mail every 10 minutes by running `mw -t 10`.

You may also want to install `pam-gnupg-git`, which can be set up to automatically unlock your GPG key on login, which will allow you avoid having put in a password to sync and send, all while keeping your password safe and encypted on your machine.

By default, mpd, the music daemon assumes that `~/Music` is your music directory. This can be changed in `~/.config/mpd/mpd.conf`. When you add music to your music folder, you may have to run `mpc up` in the terminal to update the database. mpd is controlled by ncmpcpp, which is accessible by `Mod+m`.

LARBS is deployed as a git repository in your home directory. You can use it as such to fetch, diff and merge changes from the remote repository. If you don't want to do that or don't know how to use git, you can actually just rerun the script (as root) and reinstall LARBS and it will automatically update an existing install if you select the same username. This will overwrite the original config files though, including changes you made for them, but this is an easier brute force approach that will also install any new dependencies.

You can follow links via the keyboard in this pdf reader by pressing `f` followed by the number that appears on the desired link. – For questions! – For stalking! – To incentivize more development of LARBS! – For the code behind it! – For updates!