



Published on Free Software Magazine (<http://www.freesoftwaremagazine.com>)

## Hotwire: a combined terminal/GUI for GNU/Linux

### A suitable peacemaker between command-line purists and pragmatists?

By Gary Richmond

There is nothing more guaranteed to ignite a bad tempered, incandescent flame war than an outbreak of hostilities between the rival Gnome and KDE camps. Well, except perhaps a slanging match between the champions of the GUI and the command line. Enter stage left the compromise candidate which might just unite the warring factions: Hotwire.

Whenever I first stumbled across upon Hotwire (released under the GPL 2 for the user interface and a permissive MIT-style licence for everything else), certain childhood rhymes sprang irresistibly to mind: Jack of all trades, master of none, Jack Spratt could eat no fat, his wife could eat no lean, amongst them. The problem is actually trying to give a pithy and accurate definition of Hotwire without tripping up yourself, so who better to describe it than its developer, Colin Walters of Red Hat, who has described Hotwire as “an object oriented hypershell” and “a modern Python-based extensible crossplatform shell environment”. Walters’ inspirational cue was the less powerful Windows Powershell and to make Hotwire a better, rich visual interface without recourse to yet another new scripting language. It leverages Python.

Now, you are probably thinking that GNU/Linux needs another shell about as much as it needs another patent trolling predator, but you just might be wrong because what Hotwire tries to do is to unify the GUI and CLI in one powerful, unique and productive application which gives the user, both novice and advanced, the best of both worlds. It does this by using Python and GTK to render an interface which has familiar elements of a GUI but with the granular power of the command line.

## Get it. Install it

The target platform of Hotwire is primarily GNU/Linux and Gnome but it will run happily on KDE too. The major dependencies are Python version 2, Sqlite3 and Python Gnome VFS bindings; so, you will either need to upgrade to those versions either via your distro’s package manager or upgrade your distro to the next version which does support it. I installed Hotwire on Ubuntu 7.10 with no complaints about dependencies. If installation throws up any missing dependency messages then simply install them from your comfort of your package manager. As of the time of writing, version 0.700 seems to be out and it offers more bug fixes and improved features (version 0.700 once again will support Python 2.4). For a list, see the the [Hotwire changelog](#).

Although Hotwire is described as alpha software, it is very stable and I was unable to crash or freeze it. The only sense it will it can be described as alpha software is that, as even its creator admits, it does not yet match the full feature set of Bash - but it is about eighty per cent of the way there and every update should be inch ever nearer to feature parity.

## Binaries or source?

Despite being described as alpha software Hotwire is already available in binary form for a number of distros: Fedora 8, OpenSuse, Debian, Ubuntu, Mandriva, Archlinux and Gentoo. You can [get them here](#). Just follow the links. (Followers of PCLinuxOS can install from the repositories but the version is 0.44 which is rather out of date.) Binary is best for ease of installation and removal, as well as installing Hotwire in the Start menu, with the option to create a desktop shortcut. A Debian binary for Ubuntu (version 0.620) is available at the [Getdeb website](#). Debian packages are also available for download at [Linux App Finder](#). If you cannot get an RPM for your distro you might get some mileage using Alien to convert the Debian package.

If you are having problems with locating a binary for your distribution the solution is simply to download the source code from the above link. The version to look for is 0.620. Download to the directory of choice and extract it. Cd to where you extracted it and run the following in a console: `python ui/hotwire &` (and you could always set up an alias for this to reduce typing). The ampersand is optional—I added it to run Hotwire in the background and free up the terminal for other tasks. That’s it. Extract and run the Python script. No `./configure`, `make` and `make install` and if you decide you don’t like it, uninstalling is as simple as deleting the extracted folder. For binary installs, removal is either by the GUI of the package manager or `apt-get remove hotwire`.

## First run

When you run Hotwire for the first time the default screen opens with some helpful information:

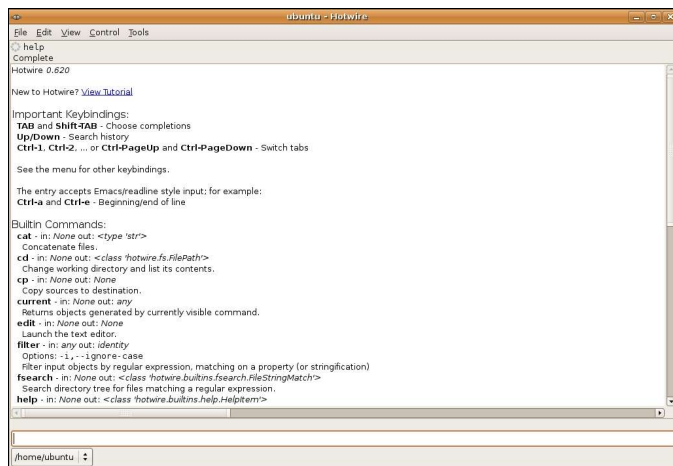


Figure 1: Hotwire default screen

It doesn’t look like any terminal/console you’ve ever seen before but you’ll soon feel right at home as soon as you type something in the box at the bottom of the screen. Let’s have a look at the contents of `etc`. Just type `cd /etc` and hit the return key. You’re in for your first pleasant surprise. Normally, that command will simply change your directory and if you want to see the contents you would have to follow it up with `ls` (and any additional arguments). With Hotwire you have killed two birds with one stone. The `cd` command not only changed directory, it also automatically listed the contents of `etc`—and what’s more listed them just as you would expect in a conventional file manager complete with icons (in this case, Ubuntu defaults):

## Hotwire: a combined terminal/GUI for GNU/Linux

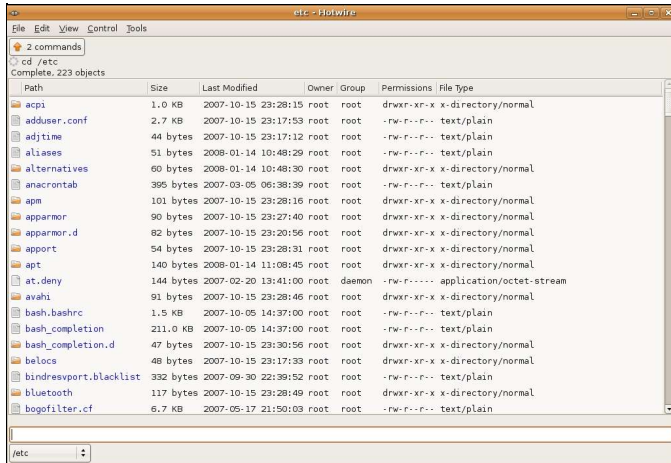


Figure 2: The results of `cd /etc` lists the contents

You might be thinking that what you see here means that Hotwire is based on XML but in fact it based on “objects”. In a thoughtful touch for the beginner, Hotwire helpfully arranges titles for the command output by categories like owners, permissions and size etc. Some of this is also available in a console but in a less transparent fashion and with less information. Like a file manager, right clicking on a file will pop up a contextual menu for viewing or editing. The nearest I have seen to this is when Konqueror is in file-manager mode and showing a horizontally-split screen with a terminal emulator. However, there the resemblance breaks down because while `cd` in the terminal follows what you click on in the main Konqueror screen (albeit an excellent way to navigate around the the GNU/Linux file system if you hate typing and can’t always remember the exact nomenclature of the commands) it doesn’t work the other way around.

## Command history

“Conventional” consoles running in X Windows have the ability to remember the last five hundred commands (see my article on [getting the most out of the history command](#)). Hotwire too can remember commands and does so via an SQL database. Unlike the usual culprits, it is not necessary to actually issue the `history` command. As you can see from the screenshot above the commands history is highly available and can be reprised by either selecting one from the top of the screen or by scrolling through them from the menu at the bottom. Nice, but there’s more. In the View drop-down menu, if you select Overview you will see a complete list of all the commands you have use in the current session and each one represents a kind of hyperlink to that command:

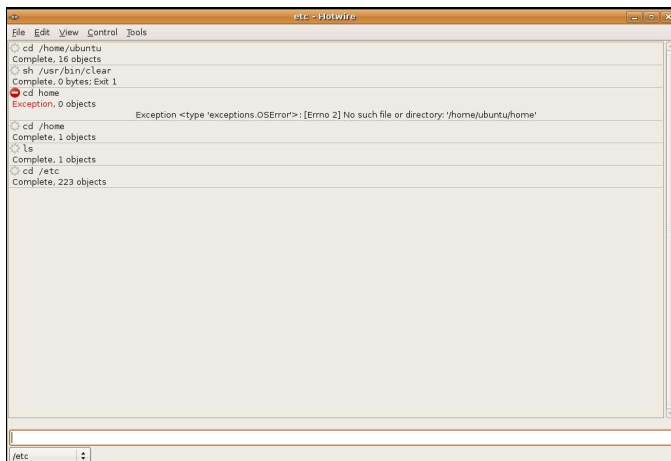


Figure 3: Overview screen listing command history

That's a useful feature, saving the need to remember a command or constantly type `history` or put the up arrow key into overdrive.

## Other features

### Other commands and search

The aim of Hotwire, amongst other things, is to obtain feature parity with Bash and to that end its widespread adoption will necessitate the inclusion of the mainstream Bash command set. Most familiar and frequently-used commands are currently supported and the best way to find out what works is simply to try them out in Hotwire. You will not be disappointed if you type any of the following traditional Unix workhorses: `man` (but in a separate tabbed terminal), `cp`, `mv`, `mkdir`, `grep`. The indispensably useful pipe command (`|`) now works and `rm` in hotwire is reversible with the `undo` command from the drop-down menu, presumably because files are “removed” to Trash (but without options, and only works in the files/directories area where the `rm` command was issued in the first instance). Hotwire does not discriminate between removing directories and files and therefore the use of the recursive `-r` is superfluous. A nice touch. Another nice touch is when you are using the virtual file system `proc`. Any item in the output can be right-clicked to kill it.

Devotees of the command line will be pleased to note that this utility supports Unix-like globbing. For example, `ls *.jpg` will output the same content as it would in Bash. Anyone who has encountered a piece of unrecognized hardware will have had recourse to `dmesg` and piping the output through `grep` to find specific information on that hardware. Hotwire deals with that too.

Run `dmesg` to get the output of the Kernel ring buffer and then hit `Ctrl+S` and Hotwire will pop up a search box at the bottom of the screen to allow you search the output of `dmesg`. This is Hotwire's Filter feature which does what `grep` does in a conventional console/terminal (including being case sensitive of course). Just click on the `X` or hit the escape key to close the search box. The search feature does not work when you deploy it against the output of a command that renders “iconified”, graphical directories in the main “canvas” (display) of Hotwire. You wouldn't tarry long in Hotwire if it increased your workload, so you will be pleased to know that it supports the equivalent of auto-completion which is a real time saver. Those who like to work remotely via the shell have not been forgotten either. Hotwire also includes an SSH terminal application, `hotwire-ssh`, which allows the user to connect and work over a remote connection using the SSH protocol.

## Term, Xterm and multiple tabs

If you find that any of your favourite commands don't work, the good news is that the designer has thought of that too. Want to launch a conventional console to get at those powerful commands? Simple. From within Hotwire type `term` and a console will open, ready for your input. However, you will now find yourself back in the familiar console territory, because when you `cd` to another directory, say `etc`, this is what you get:

## Hotwire: a combined terminal/GUI for GNU/Linux

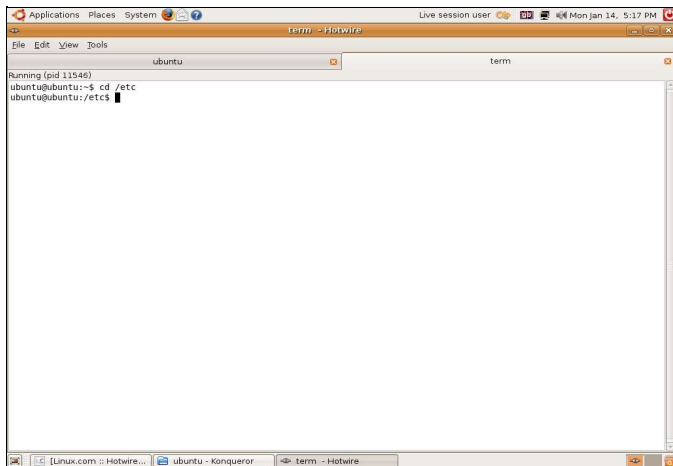


Figure 4: Term opens a tab in Hotwire

and you will only see the contents of that directory once you have issued the `ls` command. Compare this to the output when changing directory in the main Hotwire screen with `cd /etc` (Figure 2 above). If you want to work as root for any reason, Hotwire will accommodate you there too. Issuing `su` (or `sudo` for Ubuntu users) will open an interactive tab and allow you to work as root once you have entered your password. If you prefer a free-standing conventional console, type `xterm` and you are in familiar territory (because Hotwire supports embedded VTE, Virtual Terminal Emulator) and if you want to launch an application from Hotwire, follow `xterm` with the name of the application. You don't need to place an ampersand after this if you want to free up the Hotwire dialogue box and run the application in the background because Hotwire commands are asynchronous and run in the background by default.

## Conclusion

You could use Hotwire as a shoe in or replacement for your default shell, but it does not support shell scripting. This is no great handicap as you can write Python scripts instead (`py` will execute those scripts). This means that Hotwire is extensible and customizable as its designer intended it should be.

However, Hotwire can be used “as is” and it still represents a very stable and usable piece of kit—especially as the latest version is still an alpha release. Walters has described Hotwire, amongst other things, as being for both beginners and advanced users. The latter have a head start of course but the beginner has the neophyte's delight in experiencing a very graphic display (no pun intended) of the relationship between the text they type in and its graphical counterpart. It is still very much a work in progress and when it reaches feature parity with Bash I would fully expect to see it make it into the repositories of all the main GNU/Linux distros. So, Get it, install it, try it and add a fun and useful tool to the GNU/Linux armoury. You don't have to abandon the default shell. It's not an “either or” choice and it is certainly better than a “demo toy” as its creator once described it. You can use them both. It might offend command-line hardcore purists and disappoint some Python enthusiasts but you should Hotwire your distro for a quick get away. You won't regret it.

## Biography

[Gary Richmond](#) (/user/3653" title="View user profile.):

An aspiring wanabee--geek whose background is a B.A.(hons) and an M.Phil in seventeenth-century English, twenty five years in local government and recently semi-retired to enjoy my ill-gotten gains.

## Copyright information

---

**Source URL:**

[http://www.freesoftwaremagazine.com/articles/hotwire\\_a\\_combined\\_terminal\\_GUI\\_for\\_GNU\\_Linux](http://www.freesoftwaremagazine.com/articles/hotwire_a_combined_terminal_GUI_for_GNU_Linux)

---