
hecate Documentation

Release latest

June 01, 2015

1	Usage	3
2	Frequently Anticipated Questions	5
2.1	Are you serious?	5
2.2	Why is it called Hecate?	5
2.3	Does it support unicode?	5
2.4	Will you support Python 2.7?	5
2.5	Why not?	5
2.6	What versions of tmux does it support?	6
2.7	How does it work?	6

Hecate is a Python 3 library for testing command line interfaces that make interesting use of the terminal, e.g. through ncurses. Unlike tools like expect or pexpect it runs a full blown terminal emulator (specifically, tmux) behind the scenes, so you can take accurate snapshots of what your application would look like when run by a real user.

Usage

Hecate is currently shall we say lightly documented (this is a euphemism for undocumented). However the tests should work tolerably well as usage examples and there are reasonable docstrings on the implementation.

Frequently Anticipated Questions

2.1 Are you serious?

Not very, no. As a concept it works and works well, and I'm fully prepared to maintain this if it proves popular, but this should be considered a semi experimental hack which just happens to be orders of magnitude better than all of the for serious mature projects you could be using instead.

On the other hand the last time I wrote a semi-experimental testing hack that just happened to be the best around (in Python), we got [Hypothesis](#), so...

2.2 Why is it called Hecate?

It seemed appropriate to name a Selenium style tool for curses based applications after a goddess whose domain includes the moon and magic.

2.3 Does it support unicode?

Please. It's 2015. Are people still writing software that doesn't? Next you'll tell me that they're writing it in a language that has raw byte sequences as its primary string type.

2.4 Will you support Python 2.7?

No.

2.5 Why not?

Because you shouldn't need it and I don't want to. Hecate scripts are stand alone applications that interact with your program through its terminal interface. You can easily test things written in any language you like, including other versions of Python.

2.6 What versions of tmux does it support?

My CI tests it on versions 1.6 through 2.0, so that's the officially supported set. It probably works on earlier ones but I haven't tested and am not very interested in doing so.

2.7 How does it work?

Behind the scenes Hecate is spawning a tmux instance and running your command with it along with a monitoring process so that we can report back the exit code (after all, who would make a testing framework that didn't report the standard mechanism for indicating errors back to you?). Hecate interacts with the tmux server using its command line tools.

I might update it to use the control protocol at some point, but it so far doesn't seem that useful to do so and that would limit the range of supported tmux versions or require a lot of version specific code.