

Path

1 What is Path?

Path is for the bash shell. It allows you to keep named versions of environment variables and switch between them easily. If you find yourself setting variables with temporary values so you can do something and then come back to what you were doing, then path is for you. If you are a developer and have a set of variables used for your build tree and the same set used for a release tree, then path is for you.

For example, lets say that `PROJECT_HOME` is the variable that points to a project home. Using path, you would run `path edit PROJECT_HOME`. An editor opens up for you to make changes. When you save and close the editor, the value is stored in a file and `PROJECT_HOME` is set to the value in the environment. If you change `PROJECT_HOME` and then want to get your saved value back, you would run `path load PROJECT_HOME` and the default value (which you edited) would be loaded into the environment.

Now lets say you want to keep two versions of `PROJECT_HOME`. Each version of a variable has a name. When we first edited `PROJECT_HOME` we didn't specify a name, so the name "default" is used. Lets create another version of `PROJECT_HOME` for a developer build tree. Run `path edit PROJECT_HOME build`. Edit and save a new value. That new value is loaded into the environment. If we want to switch between the values, we can run `path load PROJECT_HOME {config}`, where config, in this case, is either "default" or "build".

2 What's with the name?

Path started out as a `PATH` variable editor, and the name stuck. Its grown since then. Support for other variables, and configurations was added. Paths strength remains editing path-like variables.

3 What shells does it work with?

Path runs in bash. Its implemented as a set of bash functions, a set of command completion rules and a perl script. The shell functions handle the command line interface and environment while the perl script manages the contents of the files. If you are interested in porting path to other shells, let me know by [email](#). Path could be great for managing

variables between shells.