

[quotes](#) | [docs](#) | [repo](#) | [golang](#) | [sam](#) | [man](#) | [acme](#) | [Glenda](#) | [9times](#) | [harmful](#) | [9P](#) | [cat-v.org](#)

Related sites: | [site updates](#) | [site map](#) |

ACME

- [> faq](#)
- [> keyboard](#)
- [> mouse](#)
- [» readme](#)
- [> scripts/](#)

The Acme Readme

Note: This readme is meant to be open within acme but it is included here for reference, in a Plan 9 distribution you will usually find it at `/usr/glenda/readme.acme` and is usually displayed automatically when a fresh Plan 9 installation is booted, Acme-sac includes also a similar readme that also explains how to use some of its extra functionality and applications.

Welcome to acme, the editor/shell/window system hybrid. Acme is a complete environment you can use to edit, run programs, browse the file system, etc.

You can scroll the text this window by moving the mouse into the window (no clicking necessary) and typing the up and down arrows.

When you start Acme, you see several windows layered into two columns. Above each window, you can see a “tag line” (in blue). The first thing to notice is that all the text you see is just that: text. You can edit anything at will.

For example, in the left column is a directory window. If you look at the window’s tag line, you will see that it contains

```
/usr/glenda/ Del Snarf Get | Look
```

(This might be truncated if the column is narrow.) That is just text.

Each mouse button (1, 2, 3, from left to right) does a different thing in Acme:

- Button 1 can be used to select text (press it, sweep, release it), and also to select the point where text would be inserted in the window. Use it now in your `/usr/glenda` window.
- Button 2 can be used to execute things. For example, use button 1 to type `"ls -l"` before `"lib/"` in the window showing `/usr/glenda`. Now use button 2 to select `"ls -l lib/"` (press it, select, release it). As you can see, button 2 means “execute this”.
- Button 3 can be used to get things. For example, click button 3 on `"lib/"` within the `"usr/glenda"` window. Can you see how a new window shows the contents of `"usr/glenda/lib"`? Button 3 can also be used to search within the body of a window. Just click button 3 on the thing you want to search. Again, you can select something with button 1 and then use button 3 on the selection.

You can double-click with button 1 to select words; a double click at the end or beginning of a line selects the whole line. Once you have text selected, you can click on it with button 2 to execute the selected text. A single click of button 2 would execute the word clicked as a command.

Now let’s pay attention to the tag line once more. As you can see, the left part has a path. That is the name for the window and shows also the directory for the thing shown (file/directory/program output). When you

execute something using button 2, the current directory for the command is the directory shown in the left part of the tag (if the thing shown is a file, its directory is used).

As you saw before in the example, there are windows labeled `"/dir/+Errors"`, that is where Acme shows the output of a command executed in `"/dir"`.

Another thing you can see is that tag lines contain words like “New”, “Del”, “Snarf”, etc. Those are commands understood (implemented) by Acme. When you request execution of one of them, Acme does the job. For example, click with button 2 on “Del” in the `"/usr/glenda/+Errors"` window: it’s gone.

The commands shown by Acme are just text and by no means special. Try to type “Del” within the body of the window `"/usr/glenda"`, and then click (button-2) on it.

These are some commands understood by Acme:

- Newcol: create a new column of windows
- Delcol: delete a column
- New: create a new window (edit it’s tag to be a file name and you would be creating a new file; you would need to click on “Put” to put the file in the file system).
- Put: write the body to disk. The file is the one named in the tag.
- Get: refresh the body (e.g. if it’s a directory, reread it and show it).
- Snarf: What other window systems call “Copy”.
- Paste: Can you guess it?
- Exit: exit acme

Acme likes to place new windows itself. If you prefer to change the layout of a window, you only need to drag the layout box at the left of the tag line and drop it somewhere else. The point where you drop it selects the column where the window is to be placed now, as well as the line where the window should start. You can also click the layout box to enlarge its window a small amount (button 1), as much as possible without obscuring other tag lines in the column (button 2), and to fill the whole column (button 3). You can get your other windows back by button-1- or button-2-clicking the layout box.

This is mostly what you need to get started with Acme. You are missing a very useful feature: using combinations (chords) of mouse buttons to do things. You can cut, paste, snarf, and pass arguments to programs using these mouse chords. You can read this in the `acme(1)` manual page, but it’s actually extremely simple: Select a region with button 1 but don’t release the button. Now clicking button 2 deletes the selected text (putting it into the snarf buffer); clicking button 3 replaces the selected text with the snarf buffer. That’s it!

For more information, read </sys/doc/acme/acme.ps> (you can just button-3 click on that string to view the file).

[Powered by werc](#)