Dennis Ritchie

Ben Klemens

13 October 2011

[I'm interrupting my series of semi-daily tips on C and UNIX because Dennis Ritchie, the original lead author of C and UNIX, has died[1].]

I've been reading the UNIX-Hater's Handbook[2] [PDF book] over the last few days. It's a compilation from the mid-1990s of cathartic comments from all sorts of people who see the flaws and illogic in UNIX-like systems. There's no central authority maintaining control, so there's nobody to make sure all the little utilities have a consistent interface. Shell builtins don't have `man` pages—and for that matter, `man` sucks. Et cetera. We should all be using MS DOS or Mac OS 4, which are so much better that they will some day overcome UNIX.

Reading a book from back then is funny now because so many of the complaints don't really matter anymore. The documentation is fixed, much effort has been made to standardize how command lines get parsed, and the windowing systems are beautiful. And UNIX won—there's only one operating system I know of that competes with it.

You may have heard that the CEO of Apple died last week. He worked partly on hardware, but also a great deal on software that was originally from NeXT, a system that didn't get too far past the universities, which provided a beautiful front end to UNIX, leaning on Objective C for coding up the windows. When the world gave up on NeXT and the CEO came back to Apple, the software became Mac OS X. Later, Apple put out another variant of the operating system for telephones. So Apple owes it all to Ritchie et al.

As for C, it wasn't the first language at its level, but it was the one that got enough right that we're still able to use it 40 years later, in our own style. Fortran is fighting to remain relevant, COBOL is a sort of running gag, but there's C, still the basis for what is probably most of the software you'll deal with today. Every language that isn't C is at some point in the documentation explained with a sentence beginning with *it's like C, except that. . . .*

What I really get from the story here is that Dennis Ritchie did something that improved upon what he had as well as he could, and then when it was good enough, let go. I'm writing C in my hypermodernist style, the Linux kernel hackers are writing in a totally different style, and they are producing a product largely compatible with Apples and Sun boxes, but we don't see Ritchie on stage anywhere lecturing about his vision for the system and how it will only work if he remains the center of development focus.

[1] `http://arstechnica.com/business/news/2011/10/dennis-ritchie-the-giant-whose-shoulders-we-stand-o`
`ars`

[2] `http://simson.net/ref/ugh.pdf`

His letting go of a good thing was a key contribution, and is part of why UNIX-type systems and C are still the reference for of all of modern computing.