Re: [Mu] colors in c; looking backwards

From: "Alfie Costa" <agcosta@gis.net>
To: mulinux-forum@lists.sourceforge.net
Date: 2005-02-22 23:59

On 21 Feb 2005 at 18:13, Michele Andreoli <mulinux-forum@lists.sourceforge.net> wrote:

> The question is: why to use mu, today? This is my astonishemnt,
> because I myself do not use floppy-linuxes anymore.

Oy Michele, that's like asking why study history? Why watch a silent movie? A black & white movie? People return to old stuff when the new stuff is missing something, or when they're curious about the past the better to foresee how future things may happen.

Here's some new rehashes of older stuff that have no logical reason to exist if the past is uninteresting. I'm not native to any of it -- I just stumbled into these links over the years and admire the amateur spirit:

CP/M Main Page
Last updated: 1 January 2005
* CP/M links
* CP/M information
* CP/M software
* ZINC - clones of some CP/M 3 utilities

http://www.seasip.demon.co.uk/Cpm/index.html

Jeff Vavasour's TRS-80 EMULATION Page
Updated December 05, 2004.

"If your browser supports Java, you'll be able to write a BASIC program directly in your browser! This is an adaptation of my earlier TRS-80 Level I BASIC Simulator into Java. This Java version (in theory) will run on any computer in any browser that supports Java and JavaScript. As it is not an emulator, but rather a recreation, I took the opportunity to add a couple additional features including colour support.

"I did this project for several reasons. Computers these days have made programming inaccessible to the average person. I wanted to reclaim the "old ways" a bit, where a person could tinker, learn and share what they've found. I chose a simple BASIC like Level I BASIC because I wanted something easy for people to learn."

http://www.vavasour.ca/jeff/trs80.html
(The same site also has an online java Timex-Sinclair 1000 emulator.)

TRS-80 Tandy Color Computer
Ring Manager: tlindner Join This Ring
"A web-ring connection for the CoCo fanatics, who just never seemed to grow up and realize that technology has passed them by. Or maybe we just like living in the colorful beginnings of the personal computer era, where everything was fun, with dinky little sounds and blotchy blurry graphics. Where programming was simple enough to stay fun yet challenging enough to keep one busy for hours..."

http://m.webring.com/hub?ring=coco

Such things make no sense from a certain kind of "practical" perspective, but pondering why they exist gives perspective on why the 'mu' list evokes good will.

Perhaps it's similar to the appeal of restricted programming puzzles that border on masochism:

The INTERCAL Resources Page   21 Jun 2004

OK. You've coded in C. You've hacked in LISP. Fortran and BASIC hold no terrors for you. You write Emacs modes for fun. You eat assemblers for breakfast. You're fluent in half a dozen languages nobody but a handful of übergeeks have ever heard of. You grok TECO. Possibly you even know <shudder> COBOL.

Maybe you're ready for the ultimate challenge...INTERCAL.

INTERCAL. The language designed to be Turing-complete but as fundamentally unlike any existing language as possible. Expressions that look like line noise. Control constructs that will make you gasp, make you laugh, and possibly make you hurl. Data structures? We don't need no steenking data structures!

http://www.catb.org/~esr/intercal/

And so forth. ('Forth', the computer language, also has remarkable amateurs.) These things might seem like hopeless dead ends, but in a way they're more like unexplored lands or uncataloged species. Popular interest has shied away from them, but maybe there's some special little thing, (or BIG thing), trick or talent these topics embody that makes them worthwhile.

What's 'mu' got, (or what did it have)? Small, script centric, good fractured docs, a decent mailing list, humor... for a while it had "state of the art" rustic hardware detection -- the big distros had years of lousy hardware detection, (which then got better).

What befell 'mu'? Who can say -- it was mostly a one man show. For the users: little attritions, mailing list spam problems, lack of time, "competition" in the form of similar things that had advantages in some respects, and that the structure of the "project"(???) didn't "scale" well -- no packaging or BTS, maybe a bit too informal about GPL customs, (the Tomsrbt guy used to be testy about that), etc.

Again, the available Linux Distros often have great stuff, but I think they're missing certain ingredients which attempting to describe would make this message longer than time permits.

(Come to think of it though, I haven't actually used 'mu' much lately either. Is defending it hypocrisy?)