

The Embedded Muse 87

Editor: Jack Ganssle (jack@ganssle.com)

October 30, 2003

You may redistribute this newsletter for noncommercial purposes. For commercial use contact info@ganssle.com.

EDITOR: Jack Ganssle, jack@ganssle.com

CONTENTS:

- Editor's Notes
- Toolchains Aren't Forever
- Salaries
- Jobs
- Joke for the Week
- About The Embedded Muse

Editor's Notes

I'll present my Better Firmware Faster seminar December 5 in San Jose, the last public class this year. See <http://www.ganssle.com/classes.htm> for more details. This is the only non-vendor class around that shows practical, hard-hitting ways to get your products out much faster with fewer bugs.

There's also cheap fly-in options listed there for folks coming from out-of-town.

I often do this seminar on-site, for companies with a dozen or more embedded folks who'd like to learn more efficient ways to build firmware. See <http://www.ganssle.com/onsite.htm>.

Need to build a great watchdog timer? Most WDTs are afterthoughts that aren't too likely to save a crashed system. Check out the paper "Great Watchdogs" at <http://www.ganssle.com>. And thanks to so many of you who contributed ideas and thoughts on this subject.

One of the reasons for the success of Apple's Mac and for the fanatical devotion of its users is the common user interface. Every program works pretty much as you'd expect, regardless of vendor. That's partly because of Apple's insistence on a standardized user interface. Now Unix has something similar. Eric Steven Raymond has published a book-

Copyright 2002 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at info@ganssle.com for more information.

length piece called “The Art of Unix Programming”. It’s a delightful set of rules and observations about developing Unix code.

One quote: “To do the Unix philosophy right, you have to be loyal to excellence. You have to believe that software design is a craft worth all the intelligence, creativity, and passion you can muster. Otherwise you won’t look past the easy, stereotyped ways of approaching design and implementation; you’ll rush into coding when you should be thinking. You’ll carelessly complicate when you should be relentlessly simplifying — and then you’ll wonder why your code bloats and debugging is so hard.”

Much wisdom there, indeed. Even if you’re not a Unix developer, check it out at <http://catb.org/~esr/writings/taoup/html/>.

Toolchains Aren’t Forever

Embedded systems have incredibly long lives. Some stay in service for decades. How do we maintain these devices for such a long time? Bob Dowling wrote with some questions on this; it’s an interesting and troublesome problem.

There are a couple of dimensions to consider. First is the CPU. It seems to me that a long-lived product cannot use novel architectures. Pick a cool new CPU that might not be around in a few years and your tool support will disappear as well. It’s interesting that a few families have – so far – survived the test of time. 8051, 68k, PIC, and a couple of others have been around forever.

The second issue is way one uses the development language. Plan on the compiler vendor going out of business, and the CDs you’ve stuck in the safe no longer running on Windows 2013. (Of course, no one will be able to read a CD in 2013, just as we can’t read 8” floppies today.) The new compiler you buy won’t know about language extensions supported by the defunct one. So write your code using plain vanilla ANSI-standard constructs. Obey warnings from the compiler and Lint.

Leave the code and all development scripts in an active version control system. The VCS database gets moved from server to server over the years, but stays viable. Any offline media (DVD, CD, tape, floppy) becomes obsolete too rapidly.

Another, tricky, problem is preserving the development tools themselves. I know lots of companies that put the entire development system – PC, emulators, ROM burners – into a locked closet, ready for use when a software change is needed. That works over the short term, but electronic products do go bad with age. Capacitors leak, magnetic memory decays, and insulation rots. It’s naïve to expect a complex electronic device to work when resurrected years later.

Copyright 2002 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at info@ganssle.com for more information.

We can assume that future maintenance people will replace the development computer from time to time. The PC of tomorrow will run at 80 zillion gigaweeneihertz, and use optical cube memory blogs, but will hopefully offer enough backwards compatibility so it's possible to port the code. But emulators and the other hardware tools we use may go out of existence. How will the future-person download and test code in the target system?

Perhaps this suggests that even if you use an emulator or other complex tool it's important to also have a dead-simple debug port, like a ROM monitor or BDM/JTAG interface.

What do you think?

Salaries

Software Development magazine has just released the results of their latest salary survey.

It's online at:

<http://www.sdmagazine.com/documents/s=8939/sdm0311a/sdm0311a.html?temp=RdNwy-oCyH>

The percentage of respondents who claimed to be in the embedded world continues to decline for the third year in a row. This may be due to the magazine's demographics... or perhaps to more outsourcing of development. Almost 11% of respondents are looking for work because their job was outsourced, the first year this factor has been even measurable. Only half as many managers are losing their jobs for this reason.

29% report losing their jobs because of corporate downsizing. Yikes.

Salaries haven't changed much in the last year (d'oh, you say, looking at your pay stub). Median and mean salaries for non-managers remains at \$77k and \$79k respectively. The lowest 25% average \$64k compared to \$92k for the highest quartile.

Staff bonuses run about \$5k... but few of the embedded folks I talk to report much in the way of a bonus.

Indicative of the abysmal state of the market, only 36% have been contacted by headhunters in 2003, compared to 69% in 2000. Jobs are scarce.

The industry remains battered with few openings reported.

Copyright 2002 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at info@ganssle.com for more information.

Jobs

ExoTech R&D, Inc. near Boulder CO is looking for an embedded software engineer. Details at <http://www.exotech.com/id25.htm>.

Let me know if you're hiring firmware or embedded designers. I'll continue to run ads for embedded developers as long as the job situation stays in the dumper.

Joke for the Week

Is Microsoft releasing a version of Linux? Check out <http://www.mslinux.org/> for some laughs. A notable quote: "MS Linux is released under the provisions of the Gates Private License, which means you can freely use this Software on a single machine without warranty after having paid the purchase price and annual renewal fees."

But be careful clicking around the site; links to latesky.net aggressively pop up undesirable windows.

About The Embedded Muse

The Embedded Muse is an occasional newsletter sent via email by Jack Ganssle. Send complaints, comments, and contributions to him at jack@ganssle.com.

To subscribe, send a message to majordomo@ganssle.com, with the words "subscribe embedded *your-email-address*" in the body. To unsubscribe, change the message to "unsubscribe embedded *your-email-address*".

The Embedded Muse is supported by The Ganssle Group, whose mission is to help embedded folks get better products to market faster. We offer seminars at your site offering hard-hitting ideas - and action - you can take now to ***improve firmware quality and decrease development time***. Contact us at info@ganssle.com for more information.

Copyright 2002 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at info@ganssle.com for more information.

The Ganssle Group, www.ganssle.com