The Embedded Muse 104

Editor: Jack Ganssle (jack@ganssle.com) Oct. 19, 2004

EDITOR: Jack Ganssle, jack@ganssle.com

CONTENTS:

Editor's Notes

Scheduling is a huge concern for developers today. Few get it right. I've added a significant section on scheduling to my one-day Better Firmware Faster class, which is offered in Boston on November 1 and in Las Vegas December 10. This is the only non-vendor class that shows practical, hard-hitting ways to get your products out much faster with fewer bugs. 80% of systems get delivered late, often hopelessly bug-ridden. It *is* possible to do better – much better.

See http://www.ganssle.com/classes.htm for more details. There's also cheap fly-in options listed on the web site for folks coming from out-of-town.

But register soon - my able assistant cleverly negotiated free rooms for the first 15 people to sign up at each venue.

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.

I'll be at Electronica/The Embedded Systems Conference in Munich from November 9 to the 11th, presenting three talks. If you're there, stop by and say "hi!"

Interesting Articles

Montavista and others are working on modifications to Linux to support “real-time” (around 100 microsecond) responses. See http://www.eet.com/showArticle.jhtml?articleID=49900458 for information. But the always cheeky Register (http://www.theregister.co.uk/2004/10/18/linux_mobile_real_time/) quotes Linus Torvalds as saying "almost nobody wants hard real time" and that it makes the OS more complex and, for conventional applications, less responsive.

I guess we embedded people, in the immortal words of the late Rodney Dangerfield, just can’t get no respect.

http://www.linuxdevices.com/news/NS6531324140.html is an interesting article on Metrowerks’ automotive-grade Linux.

Various sources (For instance, http://news.com.com/2100-1041_3-5415719.html) are reporting that emergency and law enforcement officials came knocking when a 20 inch flat-screen TV was radiating signals on the 121.5 MHz band. Turns out this is one of the two frequencies (the other being the first harmonic at 243 MHz) used by some radio beacons to yelp for help when airplanes, boats and ships experience severe difficulties. Satellites relay the signal to ground; emergency crews figure out the rough location of the peril and send rescuers. In the maritime world these devices are called Emergency Position Indicating Radio Beacons (EPIRBs).

The radio spectrum is jam-packed. See the US allocations at http://www.ntia.doc.gov/osmhome/allochrt.pdf.

Everything we build radiates to some degree. Various agencies regulate emissions, but ultimately we’re required to design equipment that’s relatively RF-quiet.

121.5 MHz EPIRBs merely emit a warble on the distress frequencies and will be phased out in a few years. Newer versions operating at 406 MHz send a serial number coded to the unit’s owner. Some even transmit GPS coordinates, speeding rescue. They’re amazing units. I know – in 1992 I sailed from England to Rhode Island, singlehanded, and had to abandon ship after a month at sea. Four hours after turning on the beacon, while still 500 miles from land, a Canadian Navy P-3 circled overhead, calling me by name on the VHF radio. Obligatory URL: http://www.ganssle.com/jack/ostar1.html.

Refactoring Functions

XP and other agile methods emphasize the importance of refactoring, or rewriting crummy code. Refactoring means modifying the code to improve its readability or structure, without changing the functionality.

This is not really a new concept; Capers Jones, Barry Boehm and others have shown that badly-written modules are much more expensive to beat into submission and maintain than ones with a beautiful structure.

Refactoring zealots demand if the code *can* be improved, we *must* rewrite it. That's going too far, in my opinion. Our job is to create a viable product in a profitable way; perfection can never be a goal that overrides all other considerations. Yet some functions are so awful they must be rewritten.

Boehm showed that the really bad functions eat 4 times more development time than the others. So it's more cost-effective to toss the bad ones and rewrite them from scratch than to try and beat the crummy version into submission. We all make mistakes; professionals recover from those errors efficiently. Besides, the second time through we know it'll be perfect.

If you're afraid to edit a function, if it breaks every time you modify a comment, then it needs to be refactored. Your finely-honed sense as a professional developer that, well, we just better leave this particular chunk of code intact because no one dares mess with it, is a signal that it's time to drop everything else and rewrite the code so it's understandable and maintainable.

The second law of thermodynamics tells us that any closed system will head to more disorder; that is, its entropy increases. A program obeys this depressing truth. Successive maintenance cycles always adds to the software's fragility, making each additional change that much more difficult. As Ron Jeffries pointed out, maintenance without refactoring increases the code's entropy by adding a "mess" factor (m) to each release. The cost to produce each release looks something like: (1+m)(1+m)(1+m)…., or (1+m)^n, where n is the number of releases. Maintenance costs grow exponentially as we grapple with more and more hacks and sloppy shortcuts. This explains that bit of programmer wisdom that infuriates management: "the program is too much of a mess to maintain".

Refactoring incurs its own cost, r. But it eliminates the mess factor, so releases cost 1+r+r+r…, which is linear.

Luke Hohmann advocates "post release entropy reduction." He recognizes that all too often we make some quick hacks to get the product out the door. These entail a maintenance cost, so it's critical we pay off the technical debt incurred in being abusive to the software. Maintenance is more than cramming in new features; it's also reducing accrued entropy.

Refactor to sharpen fuzzy logic. If the code is a convoluted mess or just not absolutely clear, rewrite it so it better demonstrates its meaning. Eliminate deeply nested loops or conditionals – no one is smart enough to understand all permutations of IFs nested 5 levels deep. Clarity leads to accuracy.

Jobs!

Let me know if you're hiring firmware or embedded designers. I'll continue to run notices for embedded developers as long as the job situation stays in the dumper. No recruiters please.

IBM developerWorks, an on-line publication covering the Power Architecture technology, is actively seeking writers for our new content area. Our focus is on POWER and PowerPC in particular, but also on chip and board design and embedded topics in general (both Power Architecture and general topics should be as vendor-neutral as possible, and at an intermediate or advanced level). We don't mind working with new authors, and are happy to answer any questions or field comments at dwpower@us.ibm.com. For more information please see http://www-128.ibm.com/developerworks/power/ and the author FAQ at http://www-106.ibm.com/developerworks/library/pa-authorfaq.html.

General Software in Bellevue, Washington is expanding its Technology Team with new Firmware and BIOS Development Engineers. In this position, you will design, develop, debug, maintain, adapt, and deploy firmware, including General Software's Embedded BIOS product and its other firmware products for x86-based platforms; and work with all aspects of firmware development, including chipset programming, SDRAM programming, microcode update, suspend to RAM, suspend to disk, develop software that implements BIOS initiatives (like SMBIOS for example), multiprocessing, memory arrays beyond 4.2 gigabytes, Flash programming, ATA100 programming, and much more. Please submit your cover letter and resume via email to Steve Jones, CTO, General Software, Inc., at jobs@gensw.com, addressing the minimum qualifications. For more information visit General Softwares web site at www.gensw.com.

Joke for the Week

In honor of the USA's elections this coming November 2, here's a fictional article I wrote two years ago about a pundit's reaction to this season's vote.

News Analysis

November 3, 2004.
By Daniel Shorrt

President-elect Bubba "the can man" Jones expressed surprise at his unexpected win in last night's quadrennial elections. With 100% of the precincts reporting in, Jones overwhelmed the incumbent by acquiring an astonishing 65.536% of the vote. His opponent garnered just 1.024%, with the rest strangely going to an as-yet unidentified candidate named "hckergrrl".

Seeming confused by the crush of reporters and well-wishers Mr. Jones continued to hover protectively over his shopping cart. Suspiciously eying members of his new Secret Service detail he was heard complaining about attempts to take his collection of aluminum cans and old clothes. "Nobody ain't gonna rip off my stuff," he muttered. "Dem cans worth 2, mebbee 3 bucks."

Hustled into the waiting limo he expressed delight at the prospect of sampling the car's mini-bar. Tonight the president-elect remains in seclusion in the Blair House, hastily-recruited aides telling reporters he was "sleeping off the effects of a trying campaign."

Though it remains unclear just how Mr. Jones won the presidential election despite his not having been on the ballot, this reporter has seldom seen an election run in such an efficient and forthright manner. Contrast last night's speedy decision with the rancorous climate of just four years ago, where slim margins and problems with paper ballots led to a court's decision that disenfranchised half the nation’s voters.

Modern technology has eliminated the age of hand-counted ballots. "Hanging chad" will be nothing more than a colorful phrase recalled in history books. Recounts now take seconds. Never again will the courts decide an election. Call 2004 "The Year of the Electronic Vote."

Thank the citizens of Florida for our electronic elections. Misplaced ballots, poorly-punched selections, and perhaps even outright corruption held the 2000 presidential race in suspense for weeks. Determined to avoid the scandals, the Sunshine State replaced thousands of antiquated manual machines with the latest of electronic vote counters. High tech touch screens instantly record each voter's decision, transmitting the results to a national database when the polls close. Recounts involve nothing more than a retransmission of the data, since we know the computers themselves are deterministic, immune from fraud, and cannot make mistakes.

Initial trials in the 2002 Florida primaries seemed less-than-promising. System crashes, locked up touch screens, and confusing instructions held the McBride/Reno contest in abeyance. Yet a winner did emerge, to tackle the incumbent in November's general election. Call 2002 a prototype of success, one that blazed a trail to this new and better form of e-Democracy that our children will inherit.

The Critics

Luddites suggest that we rely too much on software for critical systems, sometimes referring to the Navy's unfortunate Smart Ship program. Recall, though, that after the sixth fleet attacked Palau in Micronesia earlier this year, the review board showed the logic of that autonomous decision: the computer realized that Palau lacked the ability to shoot back. "This clearly shows the efficacy of a system designed to win wars," Admiral Decobocker noted, "the system created a war that it could not lose. The subsequent divide overflow that crashed the flagship's propulsion system was entirely unanticipated."

(Reports indicating the ship was overwhelmed by an unruly band of Caroline Islanders who turned it into a floating casino bar remain unconfirmed).

Others mentioned last month's recall of 600,000 Internet-ready automobiles when hackers found they could deflate the cars' tires, reprogram the engine controllers to emit constant backfires, and lock the windows up and doors closed with the heat on "high". Detroit reluctantly admitted to a buffer overflow problem but stated these were minor problems blown out of proportion by the press. Spokesman Graeme Kirchner once again urged passage of the National Anti-Hacking Act, saying "these bloody unsupervised latch-key kids run rampant through the computer networks. Parents should keep them bloody well away from the bloody home computer."

To get to the truth of software risks I interviewed Tom Thorten, lead software engineer of the voting system employed so successfully in this week's elections.

Me: "Mr. Thorton, isn't this device what you folks call a safety-critical system?"

TT: "Nah, this isn't a safety-critical app. That's for avionics and nuke plants. No one's gonna die if this thing crashes. Hey, if there's a problem with this box ya just reset and carry on. Pretty much like any chunk a code."

Me: "How can you insure the vote is tabulated correctly?"

TT: "Vote? Is that what this thing does? They just told me what to put on the LCD and how to log the data. The spec was pretty light, know what I mean?"

Me: "No."

TT: "Well, it's like any system. The boss is wrapped up with making promises to customers. Her boss is usually off testifying somewhere, and the Big Boss is still awaiting extradition from Barbados. It's pretty much up to me what this puppy does."

Me: "How was this tested?"

TT: "Oh, the usual, ya know? We pounded on it some, then Bob's kid (ya know the one? Cute little tyke?) played with it for a while. It was hell getting the ice cream outa the PCB. Then we stuck it down in Florida in 2002. Worked pretty good, what I hear. Few probs, but as soon as we get some more developers we'll work out those kinks. We figured most of those old folks wouldn't hit the screen too fast, but when they did, man that ole priority inversion thing hit bad, know what I mean?"

Me: "No. What happens to the data?"

TT: "Oh, we toss it into one of those embedded DBs, a cool relational thing. Sure wish it was reentrant; those corruption problems are killing us. Then at the end of the day we TCP it to Republican National Headquarters. Or was it the Dems? One of them, they keep changing the IP address on us. Little do they know - we stuck in a back door that lets us set a new IP at will. Saves a recompile every time they change their freakin' minds. Those dudes then dump it all into the National Database. But our box logs the data and ships it out. After that it's not my problem."

Me: "So if these things are on the net, what about security?"

TT: "Well, I guess some folks might want to hack their way in, but it's pretty unlikely. We're running Embedded Win eXtraP, the most secure Win OS ever. It's so good it keeps downloading patches and fixes; hey, the system we ship evolves to something else over the course of a year. It's like, organic, man. No one really even knows what the code base looks like now. Pretty awesome, know what I mean?"

Me: "So no one can break in?"

TT: "Nah, probably not. Well, not at least if the users had a brain. I can't figure out why our customers never seem to set up a password like the read.me tells 'em to. Half them systems are wide-open! You know users, though. It's like my cable modem, you know? I see all sorts of open systems on the cable. My kid even ripped off some lawyer's Amex numbers from a Word doc on this dude's exposed C drive. Sure am proud of that little rug rat."

Me: "In Florida some people struggled with system lockups. What happens then?"

TT: "Oh, jeez, customers complain about every little thing. Just reset it, for Pete's sake, and don't get all uptight."

Me: "But what happens to all of that data?"

TT: "Bit bucket, I guess."

Me: "Huh?"

TT: "/dev/null. You know."

Well, the arcane language of the computer professional was something we English majors didn't learn in Princeton's class of 1938. But I feel it's only proper to delegate the complexities of encryption, computers, and corporate accounting to those professionals who know these things best. Proper government oversight, as mandated in the Public Security and Trusted Computing Act of 2003, insures the nation's interest will be met. Mr. Thorten assured me that the Transportation Security Administration will start examining his code base "real soon now".

One Person, One Vote

President-elect Jones inherits a fractious constituency, divided in part by the nature of the very vote that gave him the nation's highest office. Perhaps the largest issue brought to the fore by various commentators in this post-election day is that of universal suffrage. Just what does "one person one vote" mean in a high tech age? Does your vote matter?

The 2000 presidential contest resulted in a razor-thin margin. Some contend that it makes no difference who had won that election, since the difference in votes was in the noise. Either candidate would have had the support of half of the electorate.

After the next mid-term elections a slew of contests won by the narrowest of margins raised questions only recently resolved by the Supreme Court. In a unique move the nine Justices deferred to the analysis of statisticians, who argued that uncertainty is part and parcel of every data gathering exercise. They pointed to the Gallop Polls which have long had an associated error band.

In Florida and other states an automatic recount is mandated when a contest results in only 0.5% difference or less between the candidates. The statisticians argued that such a small delta is meaningless, that no amount of recounting or runoff elections would express the will of the people with any more certainty. Justices bought this argument, resulting in the famous 3 sigma election rule: if the error does not exceed 3 sigma, why worry?

As the Chief Justice wrote in his majority opinion, "hey, this is the government. You just can't expect five 9s."

After that landmark decision the liberal press was taken to task for promulgating the idea that voting was a quaint but pointless exercise. One wag suggested, for instance, that a

Republican in Maryland (there are a few) shouldn't bother casting a vote in state-wide elections. The long and overwhelming history of Democratic successes there suggests that Democrats waste their time if they vote, and Republicans waste their spirit. To a first approximation the outcome seems preordained. The old saw "my vote negates yours" now reads "my vote counts as little as yours."

This reporter would argue that every vote counts, just as it did when the Founding Fathers brought forth this great nation more than 200 years ago. The Supreme Court's unfortunate decision did not factor in the power of technology to solve most ills. I see no reason why our scientists can't employ the same technology that so accurately guided the Mars Climate Observer to the surface of Mars, or that enabled Armstrong's triumphant Apollo 13 landing, to insure every vote counts.

Don't be misled by critics who contend that, in the absence of paper records, an e-recount is nothing more than another database download. Does a ballot whose chad-clinginess is interpreted by an army of well-intentioned but exhausted observers better represent the will of the people?

The new technology of electronic voting insures every person is properly represented. Your touch-screen selection instantly tips the balance in favor of your candidate. The margin for error, due to the supreme number-crunching power of the computer, is surely zero. I look forward to the day I can vote from home using the power of the Internet and the security inherent in the most popular operating systems.

The Rest of the Story
In other news, this reporter was heartened to see the Dow Jones sharply rebound when Red Hat (the latest addition to the esteemed ranks of the 30 industrials) released their Open Vote package. At 4096, up from yesterday's close of 2048, the Dow promises release from this long-lived recession.

This is Daniel Shorrt. Thank you.

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.