The Embedded Muse 151

Editor: Jack Ganssle   (jack@ganssle.com)   October 31, 2007

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
- The Future of Engineering
- Estimating Schedules
- Yet More on Open Spaces
- Jobs!
- Joke for the Week
- About The Embedded Muse

Editor's Notes

Did you know it IS possible to create accurate schedules? Or that most projects consume 50% of the development time in debug and test, and that it's not hard to slash that number drastically? Or that we know how to manage the quantitative relationship between complexity and bugs? Learn this and far more at my Better Firmware Faster class, presented at your facility. See http://www.ganssle.com/classes.htm .

Are you in the Seattle or Minneapolis area? I'll present the Better Firmware Faster class in Minneapolis December 5, and Seattle December 7. Sign up before Nov. 5th, and receive a $50.00 discount. Registration and other info here: http://www.ganssle.com/classes.htm . You'll earn 0.7 Continuing Education Units, learn a lot, and hopefully have a bit of fun, too.

Bill Gatliff pointed out that he has published information about implementing the Blowfish encryption algorithm here: http://www.embedded.com/columns/technicalinsights/12800442?_requestid=174738

In response to my comments on consulting, both here and on embedded.com, Vladimir Vassilevsky wrote: "I work as an independent consultant since 2001. This book was a great help when we started: Harvey Kaye. "Inside the technical consulting business." I recommend this book to anyone who is interested in the consulting as a career."

The Ganssle Group, www.ganssle.com

Numerous readers pointed out this article http://www.spectrum.ieee.org/oct07/5598 , which is an interesting analysis of recent computer problems on the space station. For those in a hurry here's the spoiler: software wasn't to blame.

The Future of Engineering

We know that US students don't learn the multiplication tables or anything about science. Or is our math and science deficit really a myth? Business Week reports that we're cranking out more science and engineering graduates than the market needs, and that our educational system is working great: http://www.itnews.com.au/News/63756,tech-pay-approaches-alltime-high.aspx

At the same time our salaries are skyrocketing: http://www.businessweek.com/smallbiz/content/oct2007/sb20071025_827398.htm?chan=top+news_top+news+index_top+story .

Yet other data, some from the IEEE and ACM, show a negative bubble in these subjects in college. Engineers are, they claim, switching to less-difficult majors in droves.

Frankly, I don't know what to believe anymore, as so many articles make so many conflicting statements. The relatively recent advent of rampant offshoring has created an enormous amount of FUD that various experts, some with their own agendas, have been quick to exploit. So we're left confused by all of the differing claims. It is clear, though, that this profession is changing rapidly. It's a global enterprise. Companies are anxious to reduce costs (except for non-performing CEOs, but that's another story). Yet the demand for embedded products continues to grow.

In the last two months I've traveled nearly 100,000 miles visiting developers all over the world. This much I know is true (to steal the title from Wally Lamb's wonderful book): The engineers I met in India are excited and engaged in their profession. As are the ones in Asia and the rest of the developing world. Some of these groups have enormous resources. A couple of engineers were somewhat apologetic when I asked about their team sizes, answering that, well, their teams are pretty small, only 400-800 people on a project. The work is flowing around the world, a trend that will only increase as the developing world develops. It won't be long before China starts providing a staggeringly-huge pool of engineers.

Yet I believe there will always be a strong demand for excellent developers in Western countries. The trick is to be excellent, to stay relevant, and to develop the additional skills that globalization will require.

Estimating Schedules

Joel Spolsky has a fascinating blog post about Evidence-Based Scheduling (EBS) at http://www.joelonsoftware.com/items/2007/10/26.html . Though it is a soft pitch for his company's product the approach is somewhat novel and interesting, using a Monte Carlo simulation to create probability distributions.

His method only works if developers carefully track actual data, logging just how much time was spent on each scheduled activity. That's really good advice, though it's rarely followed.

Data suggests the average developer is engaged on a particular project about 55% of the time (using agile approaches the number is more like 66%). Random interruptions, that meeting about the health insurance plan, and the like, consume something like 18 of the 40 hours in a week. EBS ignores this, folding those distractions into the time spent actually doing a particular scheduled item. That may work, but I prefer to track those interruptions separately. It's powerful data to give your boss to show where the time is really going.

I recently reviewed a few books about scheduling: http://embedded.com/columns/breakpoint/201803531 .

On an unrelated note, an engineer recently referred me to another of Joel's posts that I had somehow missed: http://www.joelonsoftware.com/articles/fog0000000043.html . It's Joel's test of a healthy software organization. Do read the article, but he believes we should all be able to answer "yes" to these questions:

- Do you use source control?
- Can you make a build in one step?
- Do you make daily builds?
- Do you have a bug database?
- Do you fix bugs before writing new code?
- Do you have an up-to-date schedule?
- Do you have a spec?
- Do programmers have quiet working conditions?
- Do you use the best tools money can buy?
- Do you have testers?
- Do new candidates write code during their interview?

- Do you do hallway usability testing?

I'm a bit unconvinced about the last question for most embedded apps, but these others are spot-on.

Yet More on Open Spaces

An anonymous reader wrote: "The office arrangements section really hit home today in this Muse. I have worked in several office configurations over the years including cubes and private offices. (Offices are definitely the best of those.)

"About a year ago I started working in a company that has one office for our engineering group. It is 15' x 20' and includes a desk at each corner and one in the middle of the room. The door is in the middle of the long side of the room. There are 5 of us in this office, including the boss, with a part time college intern who "swaps out" time with an engineer in the office and the lab (another very small room about 500' feet down the hall.)

"I originally had misgivings about this being a possible recipe for disaster, but am now amazed how well this works for our group. It has proven to be near the top of the work environments I have had over the last 30 years. (Yeah, the private office was probably 'nicer', but the people definitely didn't work together as well as this group.)

"Previously I never would even consider such an arrangement to be viable, but have been pleasantly surprised. Even so, there are always rumors about the company building on to the facility, and each of us talks about the "dream office" we would like to have.... my guess is it would be a disappointment."

Michael Covington wrote: "About open workspaces...I do a lot of reading and writing at big tables in the University library. I find it easy to actively ignore people's whispering and other noises, even up to the level of conversational speech, if I can see where they are and know they aren't trying to get my attention. I sit facing into the room so that every source of noise is visible.

"If the noise were behind my back or -- worse -- hidden by a partition, it would be much more distracting because I wouldn't be able to confirm, with a quick glance, that it didn't require my attention."

Todd Hayes contributed: "The discussion on open work spaces has been an interesting one to follow. I have been interested in this for years and have had several discussions

with my manager about what we thought would be best. Here are a few of my insights that you might want to add to the mix:

"The group I work in now has 3 walls for each cubicle. They provide a place to hang shelves more than they provide privacy. I found myself getting sucked into discussions other engineers were having quite often. Far more often than not, it turned out to be quite productive to debate the immediate issue with them and resolve it on the spot.

"At other times, we would ask questions and spout replies without moving from our seats. I found these interruptions hardly disruptive when I was asked simple questions. I also found the quick answer from a coworker to be extremely productive for me when I was the one doing the asking. Sometimes, a third person would chime in with a very important correction or caveat about the issue.

"When I look back over the projects I worked on throughout the years, it seems to me that the interruptions were important. Although they derailed what I was working on at that instant, they would usually head off far more work by one or more others. It is my firm belief that the openness our environment provides allowed the overall project to move ahead faster and more cohesively compared to when I was given a more private place to work. It was frustrating at times to not be able to complete a straightforward piece of code for all the interruptions, but that is ok. Other engineers were collectively able to make more progress than I lost because I took the time to help.

"I have had my fair share of crunch-time efforts where a coworker and I would routinely joke around by saying "Ah, six o'clock, now we can really start to get some work done." The joke was definitely reality based but applied to a particular context: my code. A project consists not of isolated pieces of code working correctly on their own but a collection pieces working correctly together. In my opinion, interruptions facilitate more and better communication that has greater benefit than the obvious drawback of reducing the efficiency of a particular worker. Said another way, the increased efficiency of the team as a whole outweighs the reduced efficiency of the individuals.

"Now, I don't think that the openness should be taken to an extreme. I think that the people who work closely and communicate most often should be collocated in an open or somewhat open area. They should be isolated from where water-cooler discussions occur that might be distracting without adding back to the project. They should be isolated from other teams that would be having their own discussions and debates. Walls are good, but I think they should be around the teams rather than the individuals. People also need to have their space. I want a place to hang pictures of my kids and the art they create. This adds to my satisfaction.

"I think it is possible to balance all those needs. The trick for the manager is to determine where the balance point for each team is. I suspect that each team and project will have

slightly different needs and there is no one perfect formula. I do believe that the local optimum for any team will be somewhere in the middle rather than at either end of the open/private continuum."

Jobs!

Let me know if you're hiring firmware or embedded designers. No recruiters please, and I reserve the right to edit ads to fit the format and intents of this newsletter.

Cypress Semiconductor in Chennai, India is looking for a few senior/staff applications.

Senior/Staff Application Engineer: This person will work closely with marketing and sales to build applications around Cypress products to demonstrate proof-of-concept in emerging application areas such as automotive, entertainment, navigation control and related electronics. The person will be part of a dynamic team that defines, creates and documents reference designs for Cypress PSoC/Westbridge products. Other responsibilities include mentoring junior engineers with embedded design techniques, evaluating new technologies and performing competitor product tear-downs as well as training Cypress sales and marketing and customers on new solutions created.

Contact L. Balasubramanian, Senior Manager, Applications Engineering for more information: bkll@cypress.com.

Zebra Technologies has the following position available: Firmware Engineering Manager located Vernon Hills, Illinois, to manage their Label Products Firmware Engineering Group.

Responsibilities:
Under general direction, plans, conducts and supervises firmware engineering activities with varying degrees of magnitude and scope. Applies comprehensive knowledge of firmware engineering, experimental design and execution, statistical quality engineering, reliability engineering and failure mode and effects analysis to complete some or all of the following duties.

1. Analyze reports of firmware issues or requirements from a variety of sources, both internal and external.
2. Design, write, test and debug source code.
3. Analyze specifications for new features.
4. Implement new features in accordance with specifications.
5. Flowchart and clearly comment work.

6. Produce understandable documentation of new functionality.
7. Communicate progress and problems to upper management.
8. Provide guidelines and direction to staff.
9. Prepare annual performance reviews and other management-related documents.
10. Perform other duties as assigned.

Qualifications:
5 - 10 years equivalent experience in product development and project supervision in a high volume production environment. Ability to speak and write English sufficiently to communicate with management and other team members. Ability to operate a personal computer. Knowledge of Microsoft Word and Excel. Ability to manage a staff of technical professionals.

To apply for the position and learn more about our benefits package, please visit our career center at www.zebra.com/career.

Relocation Assistant is provided

BSQUARE is currently seeking an Engineering Manager/Director of Engineering for the Products Group located in Bellevue, WA. This person will be responsible for providing technical leadership to the engineering team designing and developing groundbreaking applications and infrastructure for the Mobile and Embedded Device Market. This individual will:
Help define and guide the product development process
Lead the Products team to efficiently follow development schedules, meeting milestones and producing deliverables in the timely manner
Manage direct reports of engineers, perform yearly reviews for performance and work one on one with them for career and skill set development
Actively engage in hiring and retaining the best engineering talent to ensure organizational agility
Contribute to customer and partner engagements

Required Skills
Strong understanding of software development process and methodologies
Experience in Application Development on Windows / Windows CE / Windows Mobile (Smartphone PocketPC)
Leadership and product development skills to drive the team to success

Required Experience
7 to 10 years in Software Development experience is required
3 to 5 years experience writing embedded; Windows CE/Windows Mobile highly preferred

BS/MS CS/CE/EE or equivalent

Contact Kimberly Hawkins at kimberlyh@bsquare.com.

Joke for the Week

.
Life would be so much easier if we could just look at the source code.

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*". ". BUT - please use YOUR email address in place of "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.