The Embedded Muse 19

Editor: Jack Ganssle   (jack@ganssle.com)   March 20, 1998

Discipline? Bah Humbug!

Recently I was giving a seminar to a crowd of 30 or so embedded folks, when several in the class got into a quite animated discussion of the concept of a disciplined firmware engineering process. I was rather surprised to find a small contingent that was quite dedicated to creation of firmware by anarchy. Design? Reviews? Nah. Hack it out, see what happens, and make lots of changes during debug and integration.

By and large the firmware community is much more casual about software than business programmers. After almost half a century of programming history the COBOL/FORTAN/Desktop crowd has painfully learned reasonable ways to create code. Not that they are entirely successful, but it seems the formal practice of true software ENGINEERING is much more prevalent there than in the embedded world.

Now, this is not a rant against embedded developers. In every group I speak to there's a division between those in love with chaos, a majority who'd genuinely like to find ways to work better but don't know where to turn, and a small minority who adopt and practice effective software engineering.

DISCIPLINE is the watchword for effective firmware development. Resist the temptation to take the shortcuts that are so appealing, especially in the heat of schedule pressure.

The subject came up again in the last week or two. My most recent column in ESP discussed firmware software standards. Without a decent standard, one adhered to by the entire team, decent software is quite impossible. An avalanche of feedback from readers ran the gamut of praise to condemnation. A surprising number of replies were of the sort: "hey, I write code my own way and the hell with everyone else". Again, without DISCIPLINE firmware projects will ultimately fail. Maybe heroics will save project A, but B, C or D will collapse.

Many developers fear the fancy methodologies promoted by legions of gurus. Frankly, so do I. If you're working on projects that run millions of lines of code, a complex trademarked methodology is probably essential. Most of the rest of us, though, develop products with 10,000 to a couple of hundred thousand lines of code. There are a lot of simple, practical things we can do without buying into the latest oh-so-complex methods.

Watts Humphrey has studied many of these issues and concluded that changing an entire organization from one that practices chaotic software development to one that uses a more disciplined process is difficult. Worthwhile, for sure, but perhaps being too ambitious is the road to ruin. His book "A Discipline for Software Engineering" (1995 Addison-Wesley, ISBN 0-201-54610-8) is his vision of an alternative. He suggest that we each, individually, learn and adopt some relatively simple practices that will yield greatly improved productivity with far fewer bugs.

His book is a "must read" for all of us. It's not easy. It demands attention and practice. I do highly recommend it for thinking developers looking for ways to work smarter.

Thought for the Week

Remember When?

Computer was something on TV
From a science fiction show
A window was something you hated to clean
And ram was the cousin of a goat

An application was for employment
A program was a TV show
A cursor used profanity
A keyboard was a piano

Memory was something that you lost with age
A CD was a bank account
And if you had a 3 1/2' floppy
You hoped nobody found out

Compress was something you did to the garbage
Not something you did to a file
And if you unzipped anything in public
You'd be in jail for a while

Log on was adding wood to the fire
Hard drive was a long trip on the road
A mouse pad was where a mouse lived
And a backup happened to your commode

Cut you did with a pocket knife

The Ganssle Group, www.ganssle.com

Paste you did with glue
A web was a spider's home
And a virus was the flu

I guess I'll stick to my pad and paper
And the memory in my head
I hear nobody's been killed in a computer crash
But when it happens they wish they were dead

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.