

The Embedded Muse 12

Editor: Jack Ganssle (jack@ganssle.com)

December 12, 1997

Reuse - Bah, Humbug?

I often tell people that "Reuse is Poison". The comment immediately gets everyone's hackles up, and sometimes I have to dodge tossed tomatoes.

I do think, though, that we're largely operating under a misconception about the issue of reusing software. Though reuse is probably about the only hope of ultimately solving the software crisis, the fact is that *we don't do it!*

RTOS vendors still tell me that 80% of all real time operating systems are homemade. Huh? How can this possibly square with our apparent collective will to reuse software?

The shelves at every bookstore groan under the weight of dieting books; people buy them like mad. Yet America gets a bit heavier every year. It seems we're better at talking about losing weight than actually doing it. Is software reuse the metaphorical equivalent of dieting? We talk the talk but don't walk the walk?

Reuse is poison when we pretend it's important while not actually doing it. It's time to stop deluding ourselves and start practicing what we preach. Easy? No. Critical? For sure.

And, in the spirit of offering something worthwhile to supplement my critiques, here's a great source of code. Seriously great. Check out <http://www.snippets.org> for hundreds of routines, some of which are quite useful for many embedded applications.

Thought for the Week

New Programming Languages:

'SIMPLE' is an acronym for Sheer Idiot's Programming Linguistic Environment. This language, developed at Hanover College for Technological Misfits, was designed to make it impossible to write code with errors in it. The statements are, therefore, confined to 'begin', 'end', and 'stop'. No matter how you arrange the statements, you can't make a syntax error.

Copyright 2000 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at info@ganssle.com for more information.

Programs written in SIMPLE do nothing useful. They thus achieve the results of programs written in other languages without the tedious, frustrating process of testing and debugging.

SLOBOL is best known for the speed (or lack thereof) of its compiler. Although many compilers allow you to take a coffee break while they compile, SLOBOL compilers allow you to travel to Bolivia to pick coffee. Forty-three programmers are known to have died of boredom sitting at their terminals while waiting for a SLOBOL program to compile.

VALGOL - From its modest beginnings in Southern California's San Fernando Valley, VALGOL is enjoying a dramatic surge of popularity across the industry. VALGOL commands include 'really', 'like', 'well', and 'y*know'. Variables are assigned with the '=like' and '=totally' operators. Other operators include the California Booleans, 'fersure', and 'noway'. Repetitions of code are handled in 'for/ sure' loops. Here is a sample VALGOL program:

```
like y*know (i mean) start
if pizza=like bitchen and b=like tubular and c=like
grodytothemax
then
    for i=like 1 to oh maybe 100
do wah - (ditty)
barf(i)=totally gross(out)
sure
like bag this problem
really
like totally (y*know)
```

VALGOL is characterized by its unfriendly error messages. For example, when the user makes a syntax error, the interpreter displays the message: gag me with a spoon

LITHP - This otherwise unremarkable language is distinguished by the absence of an 's' in the character set. Programmers must substitute 'th'. Lithp is said to be useful in prothething lithth.

LAIDBACK - Historically, VALGOL is a derivative of LAIDBACK, which was developed at the (now defunct) Marin County Center for T'ai Chi, Mellowness, and Computer Programming, as an alternative to the intense atmosphere in nearby Silicon Valley.

The center was ideal for programmers who liked to soak in hot tubs while they worked. Unfortunately, few programmers could survive there for long, since the center outlawed pizza and RC Cola in favor of bean curd and Perrier. Many mourn the demise of

Copyright 2000 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at info@ganssle.com for more information.

LAIDBACK because of its reputation as a gentle and non-threatening language. For example, LAIDBACK responded to syntax errors with the message: Sorry, man, I can't deal with that

C- This language was named for the grade received by its creator when he submitted it as a project in a university graduate programming class. C- is best described as a 'low-level' programming language. In general, the language requires more C- statements than machine-code instructions to execute a given task. In this respect it is very similar to COBOL.

FIFTH - FIFTH is a precision mathematical language in which the data types refer to quantity. The data types range from CC, OUNCE, SHOT, and JIGGER to FIFTH (hence the name of the language), LITER, MAGNUM and BLOTTO. Commands refer to ingredients such as CHABLIS, CHARDONNAY, CABERNET, GIN, VERMOUTH, VODKA, SCOTCH, and WHATEVERSAROUND. The many versions of the FIFTH language reflect the sophistication and financial status of its users. Commands in the ELITE dialect include VSOP and LAFITE, while commands in the GUTTER dialect include HOOTCH and RIPPLE. The latter is a favorite of frustrated FORTH programmers who end up using this language.

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 2000 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at info@ganssle.com for more information.