The Last Byte

Bad Design Inside of You

Scott Davidson

THE PROBLEM OF cross-layer design, the subject of this issue of *IEEE Design&Test*, goes back a long way. The focus here is on cyber–physical systems, but that is just one, now major, example of the issue. Those who work on embedded systems have had this worry for ages. Back when hardware was a lot slower than it is today, it was often necessary to optimize software to meet hardware constraints.

Back in college, the assignment for my assembly language programming course was to implement the Game of Life on our PDP-1 (the one *Spacewar* was written on) and handle interrupts quickly enough so that the display did not flicker.

The question is how successfully your cross-layer design objectives can be achieved while maintaining modularity of the design. The article by Lin et al. on plug and play for automotive applications uses hardware virtualization, which allows, for instance, timing models independent of hardware details.

The simplest way of performing cross-layer design is to design everything at once, which can be done for hardware or software. In the early days of programming, you wrote your own functions and data structures, and your code had access to their internals. If you implemented a stack, you might know that you just needed to add an offset to the top stack element to get access to the next element without using pushes and pops. We did this, and it worked fine until you changed the implementation of the stack and forgot that you used this hack. A problem, object-oriented programming, was born.

If we want to see what will happen when object-oriented design principles are not followed, we just need to look inward—to our DNA. We have discovered that DNA does not directly control things, but is often an analog of spaghetti code, where proteins affect proteins until something happens.

Even worse, there is extreme reuse. If a mechanism to do one thing has evolved, the same mechanism might be adapted to do many other things, some similar and some not so similar. When something breaks, due to a mutation, odd things can happen.

Let us consider primary ciliary dyskinesia, a disorder caused by several mutations. It causes the cilia, finger-like projections that stick out from the surface of cells, to lose the ability to move. Our throats are lined with cilia which move back and forth to clean out bacteria and small particles and keep them from our lungs. The disorder leads to respiratory tract infections when these things are not blocked by the immobile cilia.

There is more. One indicator of this disorder is infertility in men. How is infertility related to the respiratory tract? It seems that the same genes producing the cilia in your throat also control the tail of the sperm. If the tails do not wiggle, the sperm do not go anywhere. Presto, infertility. So, this function is overloaded. Makes sense though, right?

There is still more. About 50% of people with this disorder have the placement of their internal organs reversed, so the heart is on the right, not the left. It seems that the cilia are important in establishing the left–right axis of your body *in utero*.

It is a debugger's nightmare, which we try to avoid using good design practices. Our genetic code has been evolving for a billion years with no rewrites or new versions. It is a wonder our bodies work at all. ■

Digital Object Identifier 10.1109/MDAT.2021.3087655
Date of current version: 28 September 2021.

Direct questions and comments about this article to Scott Davidson; davidson.scott687@gmail.com; Twitter: @scottd687.

Copublished by the IEEE CEDA, IEEE CASS, IEEE SSCS, and TTTC