Volume 7 (2007)

Interface: The Journal of Education, Community and Values

5-1-2007

The Independent Army

Chris Pruett

The Independent Army

The Independent Army

Posted on **June 1, 2007** by **Editor**

By Chris Pruett <pruett@vvisions.com>

I started to play with computers when I was about 11 years old. My school had a lab full of Apple //e machines with black-on-green displays, 5 1/2 inch floppy disk drives, and amazingly noisy dot matrix printers. I'd used computers a little bit before then, but only ever to play educational video games like Math Blaster and Number Muncher. In school, I found out that not only were there a plethora of video games that were not designed to teach you math, but that you could actually create primitive games on your own.

The old Apple machines had an operating system that supported the BASIC programming language built in. BASIC really lives up to its name: it is an extremely simple and easy-to-follow language. It was straightforward enough that my friends and I were able to write programs without really understanding anything about computers at all. The commercial games on the market at that time were so simple that we were able to approximate them without any formal training or education, and I quickly realized that trying to write my own games was more fun than playing somebody else's.

The rest of this story writes itself, of course. I spent all my time in high school sitting in front of a screen, then I went to college and got a computer science degree, and then I went and got a job in the tech industry. The vector from the Apple //e machines in middle school to my first job offer as a computer programmer is very straight and well-defined. It would probably make for a very boring TV movie.

The thing is, you could replace me for any number of people my age. Some of the details might change; my counterparts in the UK were probably cutting their teeth on Amstrad Spectrum computers instead of Apples, and one of my main influences, an environment called HyperCard, was only the weapon of choice for those of us who were hacking on Macs. But the rest of the story is probably pretty much the same, especially for males working in the game industry who are between 25 and 35. There are a huge number of programmers in the game industry with backgrounds startlingly similar to my own.

We are an independent army. We learned our skills alone at home in front of our computers on our own time, and when we entered the workforce we were surprised to find that our grassroots video game programming educations were the norm. Though there was no Internet, no TV network dedicated to our hobbies, and no academic programs designed specifically for entertainment software, game programmers who are about my age share a common lingo that is almost intrinsic because most of us have very similar backgrounds.

It is interesting is that we chose to learn programming so that we could make games. Many people go into the software engineering field because they enjoy the process of working out logical and elegant solutions to problems that programming entails. These sorts of "pure" engineers see equal thrill in all programming applications because it is logic puzzles, not the end product, that interests them.

But those of us in the game industry who started coding as kids almost universally shared the same impetus: to make video games. We were not interested in making word processors or databases because those programs were boring. We wanted to make video games, and though many of us enjoyed coding for the sake of it, it was actually just a means to an end. In fact, the game industry itself knows that we're not in it for pure logic problems: game programming pay is typically much lower than programming in other sectors even though it is highly specialized and is not taught in academia. We work in this industry because we are heavily invested in the end result, and our HR departments know that we are therefore less likely to quit and leave the industry.

In *Zen and the Art of Motorcycle Maintenance*, author Robert Pirsig calls his book a *chautauqua*, a work that is designed to simultaneously edify and entertain the reader. I think that learning how to program in order to write video games is probably a form of this concept: the end goal is so entertaining and attractive that I was willing to put far more energy into learning how to make games than anything else. I suspect that many people in the industry felt the same way.

Fast-forward twenty years and games have become big business. The kids have grown up and gone to work and are now getting paid for doing the stuff that they enjoyed as kids. The industry and the audience for games has grown, and the development landscape has changed from one- and two-person teams in the 1980s to groups of hundreds of people working on a single project. As technology has advanced per Moore's Law, the industry has suddenly realized that it desperately needs "pure" programmers who have PhDs and extensive experience solving problems that are not game-specific. In short, the game industry is one of the most recent additions to the corporate world.

While there is a new generation of kids making games in their basements, the environment is has changed. On the one hand, the invention of the Internet has made such endeavors infinitely easier than they were before, as aspiring game developers no longer have to work in a vacuum. In the place of BASIC or HyperCard, we now have web pages, Flash, and javascript. On the other hand, commercial games have now diverged dramatically from what a single person,

professional programmer or middle school student, is able to accomplish on their own. It now requires significantly more effort to make a simple approximation of the latest hit PS2 game on a home computer. But I am sure that the thrill of making your own computer game is the same now as it was when I was a kid; the chautauqua concept still applies.

So I am keenly interested in what will happen when the next independent army (or, perhaps thanks to the Internet, they may be the amazingly-well-connected army) enters the game industry a few years from now. I suspect that as programmers my age all intrinsically operate on a common wavelength, so too will this younger generation of game developers (though I am sure they will be on a very different frequency). I also think that the kids that are in middle and high school right now will approach video game development in a way that is fundamentally different than industry veterans. If they can inject some diversity and original thought into the game development process, the industry will benefit as a whole.

This entry was posted in Uncategorized by **Editor**. Bookmark the **permalink [http://bcis.pacificu.edu/interface/?p=3359]** .

ONE THOUGHT ON "THE INDEPENDENT ARMY"

cork board ideas
on **February 5, 2014 at 12:16 PM** said:

I want to to thank you for this good read!! I certainly enjoyed every little bit of it. I have you book marked to look at new stuff you post