

Best Books of 2007

Adrian Kosmaczewski

2008-01-23

I have several mantras in my life. One of them is to learn a new programming language every year. Another one is to read at least 6 technology-related books every year.

I've already talked about Erlang (and boy that was the most read article ever in the whole life of this blog! More than 1600 visits just for it!) so now it's time to discuss the greatest books I've read in 2007 (ordered by preference, from more to less):

- Transcending CSS by Andy Clarke
- Founders at Work by Jessica Livingston
- The Old New Thing by Raymond Chen
- iWoz by Steve Wozniak
- Best Software Writing by Joel Spolsky
- Eric Sink on the Business of Software by Eric Sink

Transcending CSS

My big winner for this year. It's hard not to recommend this book enough, not only to those that work in the web design industry, but also to those that design, simply put. It's a beautiful book. It's a pleasure to read. It's a surprise, a bliss and a revelation, all in one.

Andy Clarke, a well-known designer in the UK, who is also now a member of the CSS committee at the W3C, tells us that HTML can and should be semantically correct, and that you can do a lot using the standard tags already available. Stop using those <DIVs> everywhere! You can give your page a meaning to begin with, and then apply a style on top of it. You can find inspiration in everyday life, and you can make all of this a truly cross-browser experience without much effort. Forget about your HTML editor; use Notepad or TextEdit or gedit and discover a new world.

This book has a deeper meaning and *raison d'être*, too; the web technologies are getting to a point of maturity as of yet unseen. We can go beyond what we've seen so far, just sticking to standards, making meaningful designs, and caring about the user.

Absolutely enlightening.

Founders at Work

This book was one of the most hyped ones in 2007. Everyone wrote about it, starting with Paul Graham, Guy Kawasaki or Joel Spolsky. And even me!

Frankly, the book is really worth every bit of the hype that surrounds it. I love computer history books (I already own a few of them) and this one is, together with “Dealers of Lightning” the one I liked the most. The stories of how Lotus, Apple, VisiCalc, Firefox, PayPal or the BlackBerry appeared and grew are simply fascinating.

The book might have had more impact and hype in the entrepreneurial world, but I prefer to see it as a landmark history book (too). I’ve started working in the 90s, during the dot-com boom, and saw many similar patterns as those described in the book; incredible market value evaluations, products strongly marketed but born dead, and incredible stories of successes that nobody would have thought to be possible. The personal computer revolution of the 70s and the 80s has many similarities with what happened in the web revolution, and also with what happens now during the Web 2.0 hype. These are tremendous waves, that redefine the entire industry. And I think, we’re doomed to relive these again and again.

The New Old Thing

I’ve already written about this book, albeit in French :) So I’ll translate what I’ve said so far there:

I’ve just finished reading The Old New Thing. The author, Raymond Chen, worked in the Windows development team since 1995 (at least) and explains the reasons behind some decisions taken during the design of different versions of Windows, since 1985 to Vista. This book is a compilation of some of the best articles in his blog.

And frankly, it’s hard to believe.

Windows Vista still has APIs used to run DOS 1.0 applications, just for the pleasure of “ad infinitum” backwards compatibility. The names of the Win32 methods are completely cryptic, impossible to remember, but Chen justifies each and every one of these oddities by different historic reasons. The registry contains informations used to change the internal behavior of the memory manager, so that Lotus 1-2-3 version 2 for Windows (1990) could work flawlessly under Vista.

I ask myself how could M\$ allow such a book to be published! It makes me wish to never, ever develop software for Windows ever again, in any programming language. I strongly recommend this book, particularly if you have technical

knowledge about the Linux kernel! The descriptions of the internal workings of Windows are impressive, with a level of detail never seen before.

iWoz

It is hard to argue the fact that Steve Wozniak has invented the personal computer as we know it today. If you had any doubts (even after reading his interview in “Founders at Work”), this book will wipe them away completely.

It is written by Woz himself. Wait, did I say written? This is a told story, that almost becomes a legend at the end of the book. Woz is not modest about his feat; but he does not brag about it either. He talks about his parents, his marriages, his children, Steve Jobs, the Apple I and the Apple][, with sincerity, humor and ingenuity.

You do not need to be a fan of Apple to enjoy this book; you just need to use a computer, remember that your parents didn't, and ask yourself, how did all of this began?

Best Software Writing

This book holds the “I” numeral, but the second version has not yet been published at the time of this writing. This book is interesting in many ways; first of all, it is part of an overall tendency to write blog-based books. Joel Spolsky, Eric Sink, Raymond Chen and others are part of this trend; popular blog posts that become excellent books when put together. This book is a compilation of what Joel found the most interesting during 2004, published in agreement with the respective authors.

This book is also interesting for another reason: I consider 2004 to be a pivotal year in our industry. Subversion was released that year, as were Rails and Firefox and many other popular packages. Not only that, but the whole Web 2.0 trend can be seen as a rising to the public eye in that precise moment. The book does not explicitly show these trends, but there is an overall feeling on all the best writing for that year, that something was going on. The dot-com boom was finally behind, and new things could happen again.

Eric Sink on the Business of Software

Finally, a complete hands-on resource, useful to those seeking to start a software business (yeah, like me :) The author is Eric Sink, founder of SourceGear, maker of Vault, a popular version control system for Windows marketed as a drop-in replacement for SourceSafe. He talks about all the aspects of running a software company: finance, technology choices, tradeoffs, human resources, everything.

Even if the book is primarily targeted to the US market (which makes some stuff useless in other parts of the world, particularly the legal stuff) I think it is worth a read, and again, Sink's a great writer and the book is clear and concise.

And what about 2008?

This is what I've already started to read this year:

- The Cathedral and the Bazaar, by Eric S. Raymond
- Programming Erlang, by Joe Armstrong
- Prototype and script.aculo.us, by Christophe Porteneuve

Any suggestions for more books welcome! Feel free to leave me a comment below.