

## *Books*

# My Best Books of the Year 2005

**An overview by Charles Ashbacher**, Charles Ashbacher Technologies,  
Hiawatha, IA, USA, [cashbacher@yahoo.com](mailto:cashbacher@yahoo.com)

I varied my technical reading a bit more this year, reading and appreciating a few more business books. This change in focus is not unique to me; in the last two years I have read many articles in the trade journals stating that businesses are intently searching for people who possess both computer and business skills. Therefore, to increase your employment options, take the time to learn a few things about the business side of your organization.

My favorite book of the year is “Will Your Next Mistake be Fatal?: Avoiding the Chain of Mistakes That Can Destroy Your Organization”, by Robert E. Mittelstaedt Jr. The main point is that catastrophic mistakes are generally not due to one big error, but are the consequence of a series of small errors that reinforce each other. Sometimes the chain is rather long and sometimes if even one of the errors had not occurred, the failure would have been avoided. Major disasters such as the sinking of the Titanic, the meltdown at Three Mile Island, plane crashes and major business failures such as Enron are all examined in detail. The most common theme in these failures is a sense of arrogance (infallibility) on the part of the people who were in the decision loop. Fortunately, there are also some case histories of organizations that were faced with a catastrophe and yet managed to handle it with a minimum of damage.

As a software developer, I appreciated the examples of how small errors can cascade into a major failure. Anyone who has been part of a failed software project will recognize nearly all of the errors that are described in this book:

- a) Failure to question an authority figure.
- b) Not trusting hard data.
- c) Trusting “hard” data too much.
- d) Believing only the data that agrees with your beliefs.
- e) Tolerating violations of proven procedure.
- f) Lack of adequate training and experience.

The Chief Information Officer (CIO) position in major organizations has risen from a niche to a crucial position. In many organizations, it is likely the most stressful job. Such a rapid rise in stature has meant that there is no accumulated wisdom that can be drawn upon. The book, “CIO Wisdom II: More Best Practices”, edited by Phillip Laplante and

---

Cite this book review as follows: Charles Ashbacher: “My best books of the Year 2005”, in Journal of Object Technology, vol. 5, no. 1, January-February, pp159-161  
<http://www.jot.fm/books/review18>

Thomas Costello, is an attempt to fill this void. While no single book can hope to fill such a chasm, they do as well as they possibly could in 400 pages.

The range of topics is daunting, a partial list of the chapter titles includes:

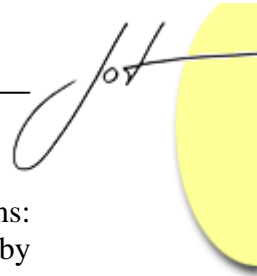
- \*) Compliance.
- \*) Securing the IT facility.
- \*) It's All About the Marketing.
- \*) Information Management: What's Next?
- \*) Software Return on Investment.
- \*) Outsource Environments.
- \*) Open Source: Time for a Plan.

There is no question that the CIO needs a broader spectrum of knowledge than anyone else in the organization.

When I mention Bernoulli trials in my basic statistics class, I note that success is whatever you define it to be. In developing software, success is largely undefined, as we generally strive for the functional rather than the perfect. With few exceptions, the software product contains errors, a fact that we must learn to live with. In his collection of essays, "The Software Development Edge: Essays on Managing Successful Projects", Joe Marasco lays down rules of thumb about how to build software, starting with how to think. Sometimes, we must be rigid in our thought patterns and at other times, it is necessary to go to Neptune and take a left. Marasco often uses narratives with his pal Roscoe Leroy, as they talk their way through the identification of a problem and some of the ways in which it can be solved. Like some of the best wisdom, it comes slightly disguised, much like the grandfather that seemed so odd to you when you were in your teen years yet whose advice you follow and crave more of after he is gone.

When you start a software project, one of the early tasks is to write out and understand the requirements. However, the general tendency is to "shoot for the moon" and create far more requirements than you can fulfill. It is here where a form of requirements triage must be implemented. The book "Just Enough Requirements Management: Where Software Development Meets Marketing", by Alan M. Davis gives you some excellent pointers on exactly where to set the bounds. By reading it, you will be able to filter out functionality when you can choose to, before circumstances dictate that you must.

As a computer science professor, I have taught two courses involving computer security and in my introductory programming classes, I try to incorporate details of how to write code with reduced vulnerabilities. Therefore, I read the book, "Secure Coding in C and C++" by Robert C. Seacord with relish. Even after years of knowing that the most common security vulnerability is the buffer overflow, it is still the most common way code is compromised. Other, less common vulnerabilities are also covered in this book, I strongly recommend that it be required reading for all people who write C and C++ code.



---

The two reference books that I found most valuable were “Core Security Patterns: Best Practices and Strategies for J2EE, Web Services and Identity Management”, by Christopher Steel, Ramesh Nagappan and Ray Lai and “The Unified Modeling Language User Guide Second Edition” by Grady Booch, James Rumbaugh and Ivar Jacobson. The need for security to be incorporated into our computer systems is obvious, and each new issue of a UML book by the three amigos is one that I will use and reuse.

### **Books mentioned in this article:**

**Will Your Next Mistake Be Fatal?: Avoiding the Chain of Mistakes That Can Destroy Your Organization**, by Robert E. Mittelstaedt, Jr., Wharton School Publishing, Upper Saddle River, NJ, 2005. ISBN 0131913646.

**The Software Development Edge: Essays On Successful Projects**, by Joe Marasco, Addison Wesley, Boston, MA, 2005. ISBN 0321321316.

**Just Enough Requirements Management: Where Software Development Meets Marketing**, by Alan M. Davis, Dorset House Publishing, New York, New York, 2005. ISBN 0932633641.

**CIO Wisdom II: More Best Practices**, by Phillip Laplante and Thomas Costello, Prentice Hall, Upper Saddle River, NJ, 2006. ISBN 0131855891.

**Core Security Patterns: Best Practices and Strategies for J2EE, Web Services, and Identity Management**, by Christopher Steel, Ramesh Nagappan and Ray Lai, Prentice Hall, Upper Saddle River, NJ, 2006. ISBN 0131463071.

**Secure Coding in C and C++**, by Robert C. Seacord, Addison Wesley, Boston, MA, 2006. ISBN 0321335724.