Automated Software Engineering Journal
Desert Island Column
Volume 10, Number 3, July 2003
pp 303-304

John Grundy
Department of Computer Science, University of Auckland
Private Bag 92019, Auckland, New Zealand
john-g@cs.auckland.ac.nz

So I'm stuck on a desert island with nothing to do but read about software engineering. To be perfectly honest I can think of a great many things that I'd be doing rather than reading or even remotely thinking about software engineering. Depending on the situation, leaving may or may not be one of them. With my wife, children, the necessities of life, and not to mention a few home comforts it could be a rather nice escape from the world of the software engineer…

But I digress. I have chosen three books, a paper and a PhD thesis. Some because I just plain enjoyed them, no matter what insights I gained from them. Some because they greatly influenced me at particular times in my research, teaching and/or practicing of software engineering. And some because they have particularly inspired me, again at certain times. The bottom line however is that all I would welcome the chance to read and re-read, most of which I do from time to time even now (desert island or no…)

Firstly, I have chosen Fred Brook's classic The Mythical Man-month. This book was basically the first software engineering text I read [OK – let's not go into the deficiencies in my undergraduate education with respect to software engineering – now that I'm Professor of Software Engineering at my alma mater I'm helping to correct them…]. I was recommended Brooks' book by one of my colleagues when working as a programmer in industry. I was working on a project that was running way over time and budget and management had decided to "hire lots more workers" to "fix the problem". We were able to disabuse them of this notion by using Brooks' work very effectively. I still think there are some real nuggets in the book for all software engineering, including those interested in Automated Software Engineering. Throwing more "agents" at the problem might not always be the best solution…

I have chosen Don Norman's The Psychology of Everyday Things next. This is a great book for anyone interested in human-computer interaction and issues for software engineers. It was a real toss-up between this and Tuft's The Visual Display of Quantitative Information (a great coffee-table book but unfortunately not likely to have one of those on my desert island…). Norman's book points out the inconsistencies and foibles of everyday objects we interact with, illustrating good (and bad) design practice. Anyone doing anything to do with user interfaces must read this book. I found it inspirational for my teaching of human-computer interaction and user interface development. I have also found it useful when working on research in the area of user interface development – often for helping to point out the deficiencies in my own interface designs! A real challenge with automated software techniques for user interfaces is preserving adequate design metrics. Norman's book will help to ground anyone getting too carried away with automation for its own sake.

I've chosen William Kent's Data and Reality: Basic Assumptions in Data Processing Reconsidered as my final book. I was recommended this book by an academic colleague while undertaking my PhD work on multiple view software engineering environments. While Kent's book focuses on databases one of the questions he asks is what is identity – particular with respect to multiple views of (database) information. I found the book a terrific read – Kent latches onto many fundamental problems with database design and technology and I found the work very applicable to multiple view-supporting systems in general. It's a very chatty style of writing that discusses real tough data representational problems and highlights deficiencies in (relational and other) modelling techniques and technologies.

I have read a great many papers over the past dozen or so years and have chosen Scott Meyer's paper Difficulties in Integrating Multiview Editing Environments as the one I'd take with me. Its quite a short paper from IEEE Software and outlines five approaches for integrating software tools with multiple views. This has been the area I've concentrated much of my research efforts over many years and Meyers' work I see as a seminal contribution, laying out very succinctly and concisely the crux of the problem and the advantages and disadvantages with common tool-based solutions. Meyers has become better-known more recently for his excellent books on C++ programming but I like this early work very much. I still recommend it to students and cite it when working in the area.

Finally, I would take with me a copy of the PhD thesis of John Vlissides, Generalized Graphical Object Editing. Again I've read a great deal of PhD theses over the years but this is definitely the best (in my humble opinion), at least in the systems/tools area. Unidraw is an OO framework supporting the development of a wide variety of graphical editing tools. The thesis is extremely well written and illustrated and an outstanding example of a "systems" thesis – I've not yet come across anything even close to its quality and readability. I read this thesis first when a PhD student myself and have used it since as a guide to writing up systems-oriented software research, particularly software tools and tool applications. Vlissides has, like Meyers, gone onto much wider fame as one of the "Gang of Four" design patterns promoters. Its also very interesting to see many of the GoF book patterns in their early forms in his PhD work.

In summary, my varied collection of software engineering texts have provided me with, at varying times, practical insight into the discipline, inspiration and motivation for my research and teaching, and a good read when wanting to forget about writing my own papers and lectures (and especially my own PhD…). It remains to be seen, however, if even they will be sufficient to persuade me to flag down that passing ship and return me to the life of an academic, or whether I'll prefer to remain in my tropical paradise…

References

Fred Brooks, *The Mythical Man-month*, 2nd Edition, Addison-Wesley, 1995.

Don Norman, *The Psychology of Everyday Things*, Basic Books, 1988.

William Kent, *Data and Reality: Basic Assumptions in Data Processing Reconsidered*; North-Holland, 1978.

Scott Meyers, *Difficulties in Integrating Multiview Editing Environments*, IEEE Software, vol. 8, no. 1, January 1991, pp. 49-57.

John Vlissides, 1990: *Generalized Graphical Object Editing*, PhD Thesis, Stanford University, CSL-TR-90-427.