File 20081113.0009: Dr Ashbourn convinced me to take all of the following out, so I'm sticking it here for safekeeping:
This is a progress report. It is not written in formal academic style. As I will show in §**??**, I can write that way when appropriate (and will do so for the dissertation and for publication), but the goal of this report is to be readable and interesting, since I am introducing you to the background of my research problem for the first time.

The purpose of this report is four-fold; namely, to convince the assessors that the following assertions are true:

- That I know how to do scientific research.
- That the problem I have chosen is an important one, that will add to the sum of human knowledge if solved.
- That I have a well-thought-out plan and methodology for attacking the problem.
- It can be done at Oxford.

I shall accomplish my purpose by showing you concrete evidence for each of the above assertions.

Now, I wouldn't want to give you the impression that just because NSA hackers have tested our system, and generally found it to be acceptably secure, that this is considered a high enough level of assurance for classified data. Not even close. You see, this is a cross-domain system. That means we get installed in a lot of 'special' locations where not just one data owner has got a dog in the fight. Usually there are several accrediting agencies involved. And the thing is, CIA doesn't trust NRO; NGA doesn't trust DISA. The Navy doesn't trust the Air Force. Most of the agencies trust NSA, but NSA doesn't trust anybody. So they all do their own CT&E. That's why we have a dedicated test lab, because the people there are kept busy, almost all the time, testing and re-testing every release of the software using different test data and different procedures in the presence of observers from every competing agency. Bugs get written up, changes go back to the developers, and the cycle repeats.

There were other problems with the evaluation, and I'll tell that story once I've figured it all out, and after I've figured out how to avoid being sued for libel.

It's important that this information be disseminated, not hidden.

You have to count the ones that didn't come back.

I don't have the budget to run multiple trials that cost that much. So let's do what we can instead: try to get something out of the sunk cost by studying what happened.

Ironically, I have never met anyone in NIAP; I was contractually prohibited from speaking to anyone there during the CC validation process. I am no longer under that restriction.

One time in early 2001, I was forced to withdraw an accepted paper from the 10th USENIX Security Symposium in Washington, DC because of an NSA request. After further review, NSA approved the paper for public release in 2002, whereupon it was published.

Doing Science

Finally, I have also spent some time reading in the literature of areas outside the exact focus of my dissertation, but related to it.

Scientific Writing

Here are good books about how to write scientific papers: [1, 2, 34, 49, 45, 54].

Recently I found some good advice on constructing arguments: [49] cites in his article a book about argument written by a US Supreme Court Justice [45] that echoes the advice I received from Ivan Fléchais: always be first to acknowledge the shortcomings of your own thesis. Every theory has holes in it[1]; the lawyer seeks to 'yield indefensible terrain—ostentatiously.' I have noted the holes in my own thesis at the end of this report.

> For a successful technology, reality must take precedence over public relations, for nature cannot be fooled. [18]

The physicist Richard Feynman (1918–1988) was justifiably famous for the clarity of his scientific writing [19, 18, 25]. Danny Hillis, who worked with Feynman on the Connection Machine, remembers:

[1] I don't have a good source for this assertion; perhaps I will find one in the course of this research.

> . . . [he] would give a sentence-by-sentence critique of the planned presentation. "Don't say 'reflected acoustic wave.' Say [echo]." Or, "Forget all that 'local minima' stuff. Just say there's a bubble caught in the crystal and you have to shake it out." Nothing made him angrier than making something simple sound complicated. [26]

The best piece of technical writing I have ever encountered was in *The Elements of Programming Style*:

> A `THEN-IF` is an early warning that a decision tree is growing the wrong way. A null `ELSE` indicates that the programmer knows that trouble lies ahead and is trying to defend against it. An `ELSE GOTO` from such a structure may leave the reader at a loss to understand how the following statement is reached. A null `THEN` or (more commonly) `THEN GOTO` usually indicates that a relational test needs to be turned around, and some set of statements made into a group with `DO-END`. [30]

Archaic programming language terminology notwithstanding, that is a clear and concise description without a single unnecessary word.

Books About Writing

Good books I have found to specifically teach the art of scientific writing include [46] and [2]. But examples of marvellous technical writing are everywhere. I particularly like:

- *Ignition! An Informal History of Liquid Rocket Propellants* [10]
- *The Soul of a New Machine* [32]
- Knuth's *Digital Typography* [**?**]
- *The C Programming Language* [31]
- *Revised*^{5} *Report on the Algorithmic Language* Scheme [28]
- *ANSI Common Lisp* [20]

I also admire the paper, 'Growing a Language' by Guy L. Steele, Jr. ([51]).

Ph. D. Advice The books I have read and continue to refer back to include [3, 35, 37, 43], and [50].

Science These are the books and essays that have taught me what I know about doing science. Some of them are books I have read over and over again: [4, 6, 10, 12, 13, 17, 19, 23, 26, 27, 41, 53, 55, 56, 57, 61].

Software Engineering And these are the books and articles about software engineering that I think everyone should read: [9, 8, 14, 15, 21, 29, 36, 38, 39, 40, 42, 52].

Scientific Ethics

> '. . . journal editors and grant reviewers rarely (if ever) require evidence that the computational equivalent of good laboratory practices have been followed. It's therefore difficult or impossible to earn points toward tenure for "going the extra mile" to turn a program that runs into one that can be trusted.' [60]

I have a strong interest in issues of scientific ethics, plagiarism, and the philosophy of science. Here are some of the books and articles I have read on the topic: [5, 7, 16, 22, 33, 44, 47, 48, 59, 11, 58].

Summary

There is a shortage of good books on CC evaluation; [24] is still the only lengthy source in the literature. What the world needs is a good how-to on CC evaluation, and I intend to write one. Much of the necessary information hides within the consulting firms and testing laboratories who charge mightily for their services; one of the goals of this research study is to drag evaluation information out into the open where it can be used more widely.

Note that this literature search is eclectic and wide ranging. I haven't read everything I've found yet; I'm just getting started on a multi-year research project and this is just a progress report at the end of the first year.

References

[1] Mike Ashby. How to write a paper. Engineering Department, University of Cambridge, Cambridge, UK, April 2005. 6th Edition.

[2] Robert Barrass. *Scientists Must Write*. Routledge Falmer, 2nd edition, 2002.

[3] Peter J. Bentley. *The PhD Application Handbook*. Open University Press, McGraw-Hill House, Shoppenhangers Road, Maidenhead, Berkshire, England SL6 2QL, 2006.

[4] Peter Bock. *Getting It Right: R&D Methods for Science and Engineering*. Academic Press, 2001.

[5] Matthieu Bouville. Crime and punishment in scientific research. *arXiv.org*, 0803(4058), 2008. `http://arxiv.org/abs/0803.4058v3`.

[6] John Brockman, editor. *What We Believe but Cannot Prove*. Harper Perennial, 2006.

[7] Ken Brodlie. Uncertainty visualisation, February 2008. Oxford e-Research Centre OeRC Seminar, 11th February 2008.

[8] Frederick P. Brooks. *The Mythical Man-Month*. Addison–Wesley Professional, 2nd edition, 1995.

[9] Fred Brooks, Jr. No silver bullet: Essence and accidents of software engineering. *IEEE Computer*, 20(4):10–19, April 1987.

[10] John D. Clark. *Ignition! An Informal History of Liquid Rocket Propellants*. Rutgers University Press, New Brunswick, New Jersey, 1972.

[11] John Coleman. CUREC rules. *Oxford Magazine*, Fifth Week, Trinity Term(277):19, 2008.

[12] H. M. Collins. *Changing Order: Replication and Induction in Scientific Practice*. University of Chicago Press, New Ed edition, 1992.

[13] H. M. Collins and R. G. Harrison. Building a TEA laser: The caprices of communication. *Social Studies of Science*, 5, November 1975.

[14] Tom DeMarco and Timothy Lister. *Peopleware: Productive Projects and Teams*. Dorset House Publishing Company, 2nd edition, 1999.

[15] Robert B. K. Dewar and Edmond Schonberg. Computer science education: Where are the software engineers of tomorrow? *Crosstalk: The Journal of Defence Software Engineering*, 21(1):28–30, January 2008.

[16] Paul Ekman. Why don't we catch liars? *Social Research*, 63(3):801–817, Fall 1996.

[17] Michael Faraday. *The Chemical History of a Candle*. Chatto & Windus, London, 1908. `http://www.gutenberg.org/files/14474/14474-8.txt`.

[18] R. P. Feynman. Personal observations on reliability of shuttle. In *Report of the Presidential Commission on the Space Shuttle* Challenger *Accident, Volume 2: Appendix F*. National Aeronautics and Space Administration, June 1986.

[19] Richard P. Feynman. The development of the space-time view of quantum electrodynamics, December 11, 1965. Nobel Lecture.

[20] Paul Graham. *ANSI Common Lisp*. Prentice-Hall, Inc., Upper Saddle River, NJ 07458 USA, 1996.

[21] Paul Graham. *Hackers and Painters: Essays on the Art of Programming*. O'Reilly & Associates, Sebastopol, California, 2004.

[22] John Grant. *Corrupted Science: Fraud, Ideology, and Politics in Science*. Artists' and Photographers' Press Ltd, 2007.

[23] Richard Hamming. You and your research, March 1986. Transcription of the Bell Communications Research Colloquium Seminar, 7 March 1986, by J. F. Kaiser, Bell Communications Research, 445 South Street, Morristown, NJ 07962-1910 USA, `jfk@bellcore.com`.

[24] Debra S. Herrmann. *Using the Common Criteria for IT Security Evaluation*. CRC Press LLC, Boca Raton, Florida, 1st edition, 2002.

[25] Tony Hey and Robin W. Allen, editors. *Feynman Lectures on Computation*. Perseus Books, 2000.

[26] W. Daniel Hillis. Richard Feynman and the Connection Machine. *Physics Today*, 42(2):78–83, February 1989.

[27] W. Kahan and Joseph D. Darcy. How Java's floating-point hurts everyone everywhere. In *ACM 1998 Workshop on Java for High-Performance Network Computing*, Stanford University, March 1998.

[28] Richard Kelsey, William Clinger, and Jonathan Rees (*eds.*). Revised^5 report on the algorithmic language Scheme, 1998.

[29] Brian W. Kernighan and Rob Pike. *The Practice of Programming*. Addison–Wesley, 1999.

[30] Brian W. Kernighan and P. J. Plaugher. *The Elements of Programming Style*. McGraw-Hill, Inc., second edition, 1978.

[31] Brian W. Kernighan and Dennis M. Ritchie. *The C Programming Language*. Prentice Hall PTR, 2nd edition, 1988.

[32] Tracy Kidder. *The Soul of a New Machine*. Avon Books, New York, 1981.

[33] David M. Levy. No time to think: Reflections on information technology and contemplative scholarship. *Ethics & Information Technology*, 9(4), 2007. to appear.

[34] Beth Luey. *Handbook for Academic Authors*. Cambridge University Press, third edition, 1995.

[35] Stephen Marshall and Nick Green. *Your PhD Companion*. How To Books, Spring Hill Road, Begbroke, Oxford OX5 1RX, United Kingdom, 2nd edition, 2007.

[36] Steve McConnell. Cargo cult software engineering. *IEEE Software*, 17(2):11–13, March/April 2000.

[37] Estelle M. Phillips and Derek S. Pugh. *How to Get a PhD*. Open University Press, fourth edition, 2005.

[38] P.J. Plaugher. *Programming on Purpose: Essays on Software Design*. P T R Prentice Hall, Englewood Cliffs, New Jersey 07632, 1993.

[39] P.J. Plaugher. *Programming on Purpose II: Essays on Software People*. PTR Prentice Hall, Englewood Cliffs, New Jersey 07632, 1993.

[40] P.J. Plaugher. *Programming on Purpose III: Essays on Software Technology*. PTR Prentice Hall, Englewood Cliffs, New Jersey 07632, 1994.

[41] George Pólya. *How to Solve It*. Penguin Books Ltd, 1990.

[42] Eric S. Raymond. *The Cathedral and the Bazaar*. O'Reilly & Associates, Sebastopol, California, 2001.

[43] Gordon Rugg and Marian Petre. *The Unwritten Rules of PhD Research*. Open University Press, 2004.

[44] Pamela Samuelson. Self-plagiarism or fair use? *Comm. ACM*, 37(8):21–25, August 1994.

[45] Antonin Scalia and Bryan A. Garner. *Making Your Case: The Art of Persuading Judges*. Thomson West, 1st edition, April 29, 2008.

[46] Robert Shoenfeld. *The Chemist's English*. VCH Verlagsgesellschaft, Weinhelm, Germany, 3rd revised edition, 1989.

[47] Loane Skene. Undertaking research in other countries: National ethico-legal barometers and international ethical consensus statements. *PLoS Med*, 4(2), Feb 2007. `http://medicine.plosjournals.org/archive/1549-1676/4/2/pdf/10.1371_journal.pmed.0040010-L.pdf`.

[48] Loane Skene. The ethics of undertaking research in other countries, 14th May 2008. St Cross College Special Ethics Seminar.

[49] Dan Slater. Scalia and Garner: 'Yield Indefensible Terrain—Ostentatiously'. *The Wall Street Journal* Law Blog, April 29, 2008.

[50] Robert V. Smith. *Graduate Research: a guide for students in the sciences*. Plenum Press, New York, second edition, 1990.

[51] Guy L. Steele Jr. Growing a language. *Higher-Order and Symbolic Computation*, 12:221–236, 1999.

[52] W. Richard Stevens. *Advanced Programming in the UNIX Environment*. Addison–Wesley Publishing Company, 1992.

[53] Norman Swindin. *Engineering Without Wheels: A Personal History*. Weidenfeld and Nicolson, 20 New Bond Street, London W1, 1962.

[54] Jeremy Walton. How to give a really good talk, 30th May 2008. special seminar.

[55] James D. Watson. *The Double Helix*. Penguin Books, second revised edition, 1999.

[56] Barry Werth. *The Billion Dollar Molecule: One Company's Quest for the Perfect Drug*. Simon & Schuster, 1994.

[57] Pepper White. *The Idea Factory: Learning to Think at MIT*. The MIT Press, 2001.

[58] Gavin Williams. The ethics of research. *Oxford Magazine*, Second Week, Trinity Term(276):1–2, 2008.

[59] Gavin P. Williams. Central University Research Ethics Committee. *Oxford Magazine*, Second Week, Trinity Term(276):3, 2008.

[60] Greg Wilson. Those who will not learn from history.... *Computing in Science and Engineering*, 10(3):5–6, May/June 2008.

[61] David Wooten. Galileo and the experimental method, 21 May 2008. Museum of the History of Science Seminar, Broad Street, Oxford.