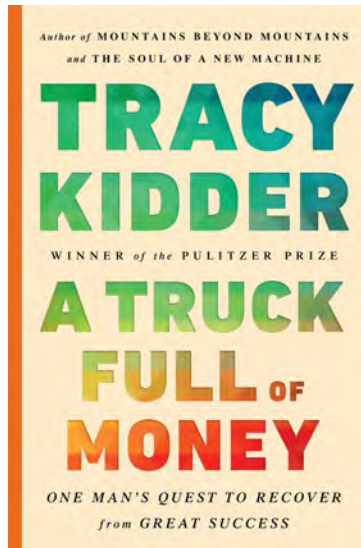


**Book review: *A Truck Full of Money*
by Tracy Kidder**

David Walden

Tracy Kidder, *A Truck Full of Money*. Random House, 2016, xxiv+260 pp. Hardcover, US\$28.00. ISBN 9780812995224.¹



In 1981 Tracy Kidder published *The Soul of a New Machine*, about the team effort to develop a new computer at Data General. Now, over thirty years and eight books later, Kidder has returned to a topic in the world of computing. *A Truck Full of Money* is primarily a profile of computer programmer and serial entrepreneur Paul English, who may be best known as one of the founders of Kayak which was sold in 2012 to Priceline for \$1.8 billion (including, as noted in the book, \$120 million for English).

The book also gives insight into the mentalities and sensibilities of various (top notch) computer programmers associated with English over the decades: how much they enjoy programming (or at least how easily it comes to them), how they think about the risk of joining a startup, their reactions to having financial success and failure, and so forth. English is especially conflicted about making money; the jacket line below the title on the front cover of the book is, “One man’s quest to recover from great success.”

Paul English, now in his early 50s, grew up in a large working class family in Boston. He was not a particularly good or well-behaved student at Boston’s renowned Latin (High) School, but he did find computers there and found that he was a natural with them. He went to the University of Massachusetts

Boston for college, taking many classes at night so he could work days to pay his way through college, and over seven years received Bachelor’s and Master’s degrees in computer science. More importantly, perhaps, he fell in with a small group of gifted computer programmers there (including Karl Berry) and faculty member Robert Morris. In the late 1980s, English got a programming job at Interleaf through Morris who worked there part time.

At Interleaf, English once again found himself in a group of like-minded people (compatible technical viewpoints), who worked well together, and were happy that English was willing to be “the manager”. He got things done, and he rose through the ranks relatively quickly to senior vice president positions of engineering and product marketing.

At Interleaf and in later entrepreneurial activities (sketched over the course of the rest of the book), English “collected” smart people, various of whom joined him in startup ventures after his time at Interleaf (and Intuit which bought Interleaf while English was there). His latest startup, Lola, was being developed as this book was being written and was announced between the time the author sent his final manuscript to the publisher and the date of the book’s publication (tinyurl.com/globekidder).

It seems to me that author Tracy Kidder likes to tell a certain kind of story in the nonfiction he writes. He focuses on a few characters and their conflicts (often internal rather than with each other or with outside forces). He likes craft, for example, building construction in *House* (1985) and computer programming in the current book. His characters are fundamentally good people, albeit with foibles, who are on a mission — people that readers, the author, and other people in the story can admire despite whatever warts they may have — people who other people want to follow, or at least put up with.

Paul English is such a character. He has an idea a minute, he makes things happen, and he can be brusque with people working for him even though he likes them and mostly he is a great boss. People in turn want to follow him, including those who tamp down some of his wildest ideas and execute his more plausible ideas. They trust him and believe they can count on him despite some of his wild or idiosyncratic ways. English is particularly conflicted about money: he makes it, he risks it, he gives it away. An important aspect of the book is the author, the reader, and English himself trying to figure out what makes him tick.

I recommend the book to readers. As is typical for Tracy Kidder, the book is easy to read with frequent

¹ A second review is at <http://walden-family.com/ieee/truckfull.pdf>.

nice turns of phrase; it features a quite fascinating character and sketches activities in domains with which the reader is perhaps not familiar—in this case the worlds of computer programming, high tech start-ups, and relatively hands-on philanthropy. The book may be particularly relevant to members of the T_EX and TUG communities: (1) Donald Knuth and T_EX are discussed several times over the course of the book; (2) Karl Berry plays an interesting role in the Paul English story told by Kidder.

More personally, I especially enjoyed the book as a one-time computer programmer myself. I got to relive through Kidder’s prose some of the joy, excitement, and feelings of changing the world that come with creating innovative computer programs and finding users for them in the world beyond the development lab.

The first Kidder book I read was *The Soul of a New Machine* (1981); it was a revelation to me in how engaging nonfiction writing about computing could be. I don’t think I read another of his books until *Mountains Beyond Mountains* (2003). Reading that took me to the library and bookstore to obtain and read five more books in his oeuvre. Now reading *A Truck Full of Money* has reminded me that I need to get my hands on *Among Schoolchildren* (1989), *Old Friends* (1993), and *Home Town* (1999).

The attendees at TUG 2014 in Portland, Oregon, had a chance to hear Tracy Kidder speak and chat with him at coffee breaks, lunches, and the banquet. (And it’s not every day that one gets to sit near, let alone talk to, a Pulitzer Prize-winning author.)

As part of his research for *A Truck Full of Money* Kidder was in contact with Karl Berry, both as someone who knew Paul English well and as someone with connections into the non-commercial part of the software world. Karl invited Kidder to attend TUG 2014, and in return Kidder graciously agreed to give a short presentation—which was primarily about his research and writing of *The Soul of a New Machine* and was highly engaging.

I have read a lot of books, but never before had the opportunity to chat with an author who is researching a book I later read. Having a little connection with Kidder during the writing of this book has led me to think about how books get written, or at least how he writes and structures books.

Kidder’s books add context (and information) to the story telling by having some chapters or sections of chapters provide background information on an aspect of the domain of the central character(s). For instance, in his book *House* there is a discussion of wood and the lumber industry.

The inclusion of Knuth in the book is a way to provide background on the world of computer programming. Kidder talks about Knuth’s assertion that approximately one in fifty people “have a peculiar way of thinking that makes them resonate with computers” and that computing projects need such “geeks” to succeed. He also recounts Knuth’s belief that programming is an art in the sense that programs can be written that are works of art. (I hope the reader will not take from this discussion the idea that programming is an art in the sense that its practitioners don’t need learned craft and discipline.) Knuth is also quoted as saying, “our pleasure is significantly enhanced when we accomplish something with limited tools.” Kidder uses the span of Knuth’s career to note that lots of programming these days is about putting existing pieces of software together instead of writing innovative new programs. Finally, Kidder uses Knuth’s parody about an “earth shaking announcement” about iT_EX at TUG 2010 in San Francisco to make the point that we now live in an app-oriented computing world. This Knuthian point of view may help readers see computer programmers as having a fascinating life when many people undoubtedly think of computers as highly bothersome (“why can’t I make it do what I want”, is a question I’ve heard) and sometimes completely infuriating (“who are the idiots that create these things”).

Kidder’s TUG 2014 presentation, chats I had with him at the conference and in a few follow-up emails, and what he wrote in *Good Prose* (with Richard Todd, 2013) brought home to me that in writing there are two voyages of discovery: first, discovering information about a topic; second, discovering a tellable story based on aspects of the discovered information. (For more tips on writing, read *Good Prose* or at least look at tinyurl.com/kidderonwriting.²)

As I ponder the content of the book, its writing, and its structure, I wonder to what extent Tracy Kidder saw parallels between his kind of non-fiction writing and the code writing of computer programmers in the book—both involving editing and rewrites, both seeking a popular audience, both highly creative. Did he come at all to identify with the computer programmers he was writing about?

◇ David Walden
walden-family.com/texland

² Also, here is an interview with Kidder about how he writes: tinyurl.com/kidder-interview (part 1), tinyurl.com/kidder-interview2 (part 2).