

Alex Groce (agroce@gmail.com), Oregon State University

Charles Babbage's *Passages From The Life Of A Philosopher* is a strange book to write about in a column reviewing classics in software engineering literature. However, in other ways it is a perfect fit: while we would have software even if Charles Babbage had never been born, and the world of computing in which we live owes relatively little to the mechanical contrivances of his 19th century, the story of Babbage's idea, and of Countess Lovelace's ideas on that idea, is in many ways a more powerful and present legend to modern computer scientists than the 20th century tale of the electronic computer. The simplicity of steampunk Dickens has advantages over a complex and murky priority-dispute overshadowed by apocalyptic war.

What makes a classic of software engineering literature? First, I'm only going to consider books; academic papers, blog posts, and computer programs are arbitrarily and summarily rejected. Second, a classic must be at least ten years old. In most fields, ten years would be a ludicrously short-sighted time in which to become a classic, but the silver-haired ancient tomes of software engineering are typically at most fifty years old, so ten years it is. Third, a classic of software engineering has to be of particular interest to software engineers, not just computer scientists in general --- there are many classics of computer science with no special appeal to SIGSOFT folk. Finally, a true classic in any literature should reward reading --- and re-reading --- and be of interest to the intelligent and perceptive reader who is not professionally devoted to the topic of the book, but has some basic knowledge of the subject. Charles Babbage's 1864 autobiography is, by this definition, a classic of software engineering.

The most read portions of the book are the chapters describing Babbage's ideas for mechanical computers. The first and second Difference Engines were great advances on previous calculating machines, but fundamentally no different than Pascal's or Leibniz's devices. However, in his Analytical Engine Babbage proposed the first workable notion for a truly general computing device, and thus the foundation for the creation of instructions for computing devices -- thus a basis for software engineering. Given Babbage's idea of "a machine of the most general nature" the immediate implication is that the instructions (for Babbage, two series of punched cards) must themselves be engineered. He touches briefly on the re-use of cards as a "library," the human factors of how operators may properly introduce cards into the machine, and even the efficiency of caching results vs. computing results from scratch. A pleasure of this book for a software engineer is to note the appearance of many of our present problems long before software existed anywhere except in the minds of a handful of men and women thinking a century ahead of themselves.

Babbage constantly touches on another topic of great interest to software engineers: notations for communicating complex designs. Chapter IX, "Of the Mechanical Notation," for example, has points which touch on the dream of an effective and useful graphical notation for the operations of software. Babbage desires to go beyond "ordinary mechanical drawings"

which only provide the “shape and relative position” of the components of a machine, to what we might think of as a conceptually executable notation. He wants to make clear “the actual time and duration of every motion throughout the action” and to capture both sequential and concurrent operations, by the use of formal rules. These rules are designed not only for a “new demonstrative science” of mechanical drawing but with an eye to usability by engineers, to “enable the attention to be more easily confined to the immediate object sought.” It is not difficult to imagine Babbage proposing an IDE for the design of his engines, a marriage of Eclipse and AutoCAD.

*Passages From The Life Of A Philosopher* is, simply, a strange book. Members of the Association for Computing Machinery are usually most familiar with the critical chapters concerning computing machinery, which are the least strange part of the book. The remainder of the book is less a coherent narrative of Babbage’s life than a series of set pieces on any topics that struck his polymath and most Victorian mind. Twenty pages of high fury against street musicians, and organ-grinders in particular, are perfect in their vituperation:

“An ancient philosopher divided all mankind into two sections, namely, fools and philosophers; and, unhappily for the race, the one cannot enjoy his whistle except at the expense of the other. I was once asked by an astute and sarcastic magistrate whether I seriously believed that a man's brain would be injured by listening to an organ; my reply was, ‘Certainly not;’ for the obvious reason that no man having brain ever listened to street musicians.”

Babbage was a man of strong opinions and many interests, and a talented writer, who describes a world both alien and familiar, and his companions in that world, from kings to blind salamanders. Reading *Passages* for its insights into software engineering is fruitful but in some ways like reading *Huckleberry Finn* only for its ideas about river navigation on a raft.

Reading Babbage is admittedly unlikely to be the revelation to a young software engineer that reading Brooks’ *The Mythical Man Month* can be. It will produce no immediate changes in working habits or software organization. Its benefits are more like those of all founding texts. First, it helps us take heart that the problems for which we cannot find a “silver bullet” did not even begin with software itself, but are possibly inherent in the design of extremely complex machines, whether the parts involved are mechanical or computational. Second, the book provides a connection between software engineering in itself and other fields, to engineering as a human enterprise existing for thousands of years before the first code. Babbage’s mind moves from volcanoes, to probability theory applied to religion, to the industrial engineering of needle-manufacturing with ease, suggesting the importance of subject expertise to the design of software for any purpose.

Finally, *Passages From The Life Of A Philosopher* can provide a continuous sense of wonder, at the idea of the automation of action, the construction of ideas that, through embodiment in brute matter, act of themselves. The late literary critic Hugh Kenner, a figure himself as

baroque and varied in interests as Babbage (Kenner wrote expertly about Chuck Jones, Buster Keaton, James Joyce, Ezra Pound, Buckminster Fuller, and Charles Babbage), relates in his own strange book, *The Counterfeiters: An Historical Comedy*, a kind of origin story for Babbage's notion of making devices for running software. The secret spans some of the earliest and last pages of *Passages From The Life Of A Philosopher*. As a young boy, Babbage saw the Silver Lady, a twelve-inch automaton, a figure of a woman with a bird on the finger of her hand, a bird that "wagged its tail, flapped its wings, and opened its beak." As a man, he purchased the Silver Lady, and devised a basis for a world of marvels, based not on endless trial and error in the construction of each automation, but on a universal description of operations for the "counterfeit" of nature. That concept, more than electronics or scientific purpose, separates the Curiosity rover exploring Mars from the Victorian marvel, and created software engineering.