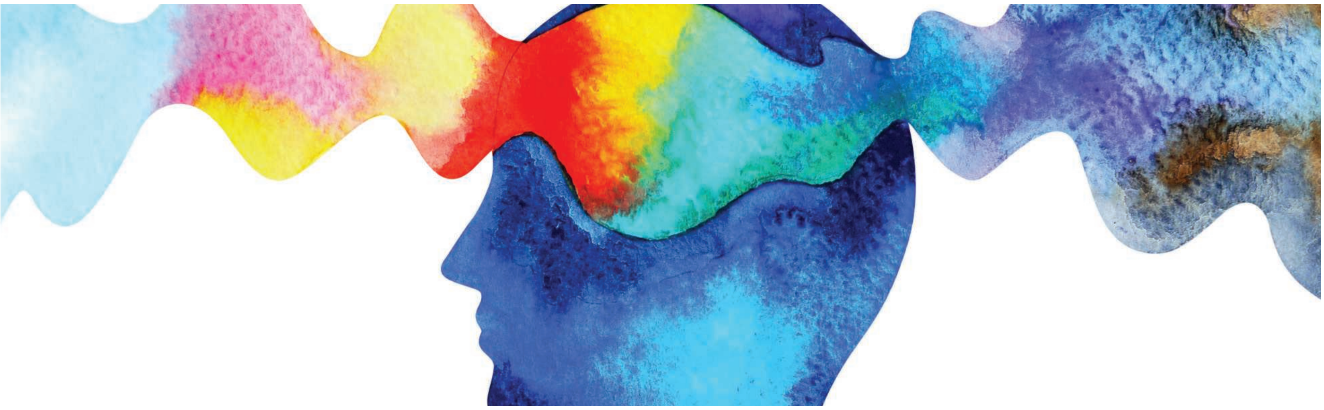

The Manual Memex

David Alan Grier, Djaghe, LLC

Here we introduce a new column that explores Computer's contribution to the computing literature and how it is laying the foundation for the future of computer science and engineering.

"Can't you make your book about the future?" an editor once asked me. "People love to buy books about the future." Her comments were well intended and forced me to rethink what I was doing. I had just completed a manuscript about the founding generation of the computer industry, the people who had joined IBM, Burroughs, General Electric, and the other computer vendors of the 1950s. Her comments were an indirect way of saying "I don't think anyone will care about your subject" and a direct way of pushing me toward a book that she thought she could sell.

As I thought about her comments, trying to avoid taking offense, I realized that I really didn't care much about the events of the 1950s and 1960s either. I was concerned with the events of today and how we would have to navigate among them. However, I was also interested in how our present world was shaped by the legacy of that founding generation, how their values and decisions shaped our landscape, and the way that we navigated across it.

Far too often, we accept a very narrow view of our field. We do not see the connections between our work and others, miss the links between seminal ideas, and subscribe to a theory of invention and innovation that simply cannot be true. Ideas do not drip uniquely on individuals, bestowing on them the power of invention and causing them to run through their offices shouting "Eureka! Eureka!" Ideas bubble through a community, taking different shapes before they settle into a useful form.

In serving as editor-in-chief of *Computer*, I regularly received manuscripts that contained echoes of familiar ideas. The authors claimed that their work was entirely original and presented evidence that they alleged to have developed, as good researchers should, without any effort to copy or borrow the work of others. Yet, a few moments of research would, generally, uncover an article from 1955, 1964, or 1987 that presented an idea almost the same as the one in the manuscript. Usually, the idea came from a different context, was expressed in an odd notation, or solved a problem in a specific subfield of the discipline. These articles were not especially hard to find, although they came from all corners

Digital Object Identifier 10.1109/MC.2019.2951852
Date of current version: 15 January 2020

EDITOR DAVID ALAN GRIER
Djaghe, LLC; grier@gwu.edu

of the computing literature: the IEEE Computer Society Digital Library and IEEE *Xplore* as well as the digital libraries of the Association for Computing Machinery (ACM), Elsevier, Oxford University Press, Springer, and the American Mathematics Society.

As I found more of those articles, I speculated that it might be useful to write a column called "Exploring Our Digital Libraries," but that idea seemed too unfocused. Few people go exploring without a clear goal, such as finding the Northwest Passage, discovering the gold mines of the Incas, or locating the Fountain of Youth. Instead, I thought it might be useful to implement a manual version of another old idea, Vannevar Bush's Memex.[1]

The Memex was a speculative device that Bush created to suggest how we might mechanize the problem of organizing knowledge, searching literatures, and identifying ideas. It has rightly been identified as part of the inspiration for the World Wide Web. But if we claim that it is a distant fount of web servers, uniform resource locators, and HTML, we slide back into that narrow world in which we can only see our work and believe that everything leads to the point where we stand today.

In fact, Bush had only a vague sense of how the Memex might work and could merely speculate about how we might build such a machine and feed the existing body of knowledge into it. However, he had a very clear sense of the problem the scientific community was facing. "The investigator is staggered by the findings and conclusions of thousands of other workers," he wrote, "conclusions which he cannot find time to grasp, much less to remember, as they appear." This problem, he noted, was caused by the trend toward specialization, of the natural tendency for workers to limit the scope of their inquiry. Specialization "becomes increasingly necessary for progress," he added, "and the effort to bridge between disciplines is correspondingly superficial."

So, in this column, I am going to try to expand the bridge between disciplines and widen the scope of our horizons. I will going to look at the some of the key contributions of *Computer* and how they have shaped our present landscape. The effort will not be as broad as I might like, and it will certainly be more manual activity than Bush proposed, but it will strive to expand our understanding of the current state of computer science, computer engineering, and related fields. It should, if I do it properly, also fulfill that goal of my early editor, which was to get me to predict the future. If we understand the current state of the field, we will be better able to peer a little way into the future.

Our starting point for this first column will be the June 2019 issue on quantum computing. It was one of the best issues that we published last year. The guest editor, Erik DeBenedictis, did a terrific job of recruiting solid articles on the subject and painting a very rich picture of the state of quantum computing: what it could and couldn't do and the likely areas for progress. The issue identified matters that may be problems for years to come, including noise and leakage as well as scaling. It argued that we should be able to see some successful specialized processors, such as the D-Wave, but that the prospects for a general quantum computer were still uncertain. If you have not read the issue, I recommend it to you.

If we are looking to outline the field of quantum computing, we jump to the article that is generally identified as the founding document, the 1985 work by David Deutsch, "Quantum Theory, the Church-Turing Principle, and the Universal Quantum Computer."[2] The article is remarkably prescient and outlines ideas that are still current in the quantum literature. But if we cling to it as the starting point of the field, we are merely backing the origin of our work. Papers on quantum computing didn't start to appear in the IEEE literature until approximately 1995 (or 1994, if you look at the ACM library).

One of these early papers is central to the quantum-computing literature. It is the article by Peter Shor on how quantum computing would be able to factor large numbers and, hence, might provide a way of compromising public-key cryptography.[3] As was common in most of the early quantum literature, the article appeared in a specialized publication rather than *Computer*. The specialized publications tend to get the earliest papers in any field, and they are edited by more coherent communities that can best understand and evaluate new work. By its nature, *Computer* is a general-purpose professional publication. It is a member benefit to individuals who belong to the IEEE Computer Society, and, hence, its content needs to be accessible to a wide range of professionals.

Perhaps *Computer*'s most important early paper on quantum is the 2002 article "A Practical Architecture for Reliable Quantum Computers," by Marc Oskin, Frederic Chong, and Isaac Chuang.[4] To date, it has been cited by 60 papers, 19 articles in IEEE publications, and 41 articles from other publishers. In addition, it has been cited by three patent applications.

Now, this is where we embrace the manual aspect of our work and move away from Bush's efforts to mechanize the organization of scientific literature. We have created many ways of quantifying the importance of a single article, and all of them are problematic. For example, one of the most highly cited articles of the past half century is Martin Fleischmann and Stanley Pons's "Electrochemically Induced Nuclear Fusion of Deuterium."[5]

That paper claimed to demonstrate that hydrogen atoms could be fused into helium at room temperature, an assertion that was quickly proven to be false. Hence, we will be misled if we read the large number of citations as evidence that the paper is important within the field of physics. It has been important but not in the way that the number of citations might suggest.

The citations for "A Practical Architecture for Reliable Quantum Computers" show that the ideas from the article moved quickly into the computer-architecture community. Most of the citations are from conferences, which tend to be populated by active researchers. The citations suggest that the article opened a field of inquiry. In general, papers do one of four things. They open fields of research, redirect their fields, combine two fields, and close fields. Those that open fields articulate a series of ideas, concepts, and processes and, then, show how those elements can be used to solve problems.

The lead author, Mark Oskin, reported that the field "was still novel" when he wrote the article. At the time, he was a young graduate student at the University of California, Davis, "more or less finished with my Ph.D. research but waiting for the academic job cycle to start." He had seen an early quantum device, a bulk-spin computer built by Isaac Chuang from IBM. Chuang was a friend of his Oskin's advisor, Fred Chong. The three would become coauthors of the article.

Their work focused on the architecture that would lead to reliable quantum computers. "The nonlocalized properties of quantum states," the paper noted, "means that localized errors on a few qubits can have a global impact on the exponentially large state space of many qubits." The nature of this problem suggested that error correction needed to be handled on a system-wide level. "Unlike classical systems, which can perform brute-force, signal-level restoration error correction in every transistor, quantum-state error correction requires a subtle, complex strategy." The article concludes, "While theoretically possible, quantum error correction introduces overheads yet unheard of in the classical domain."

For the most part, the article seems to have had a positive reception. Oskin recalled that much "to our surprise and delight, it inspired other researchers to take up the topic." The fundamental lesson of the article was that "quantum error correction has enormous overheads when implemented in practice." It wasn't going to have the capacity to break public-key codes "anytime soon."

Oskin did acknowledge that a few reviewers were not quite able to make sense of the ideas in the article. He remembered that a reviewer from one of the top architecture journals described it as "mediocre science fiction, at best." Reviewers can regularly miss important ideas, especially when a field is young. The three coauthors took the criticism in stride. "We found the review so hilarious we had T-shirts made with that quote on it."

We limit our understanding of early papers if we just view them as merely defining a basic set of ideas used by subsequent authors. If we look at the papers that cite "A Practical Architecture for Reliable Quantum Computers," we can see the subtle influence that early papers wield. They give ideas that can be used to solve problems, and they suggest problems that might be interesting to attack. "If you look at the recent focus," Oskin observed, "it has been on algorithms that do not rely on error correction and technologies that may be more naturally error tolerant and/or with lower overhead." Such algorithms are part of the substantial progress that architecture has made in recent years.

"A Practical Architecture for Reliable Quantum Computers" is part of the body of knowledge that *Computer* has helped create. That corpus has a distant foundation in a physics journal, spreads across several architecture conferences, has important contributions in processor and microelectronics journals, includes several mathematics papers, and even has a connection to *Acta Astronautica*, a journal for articles about space exploration. The digital libraries can guide us through this literature and provide some of the mechanical connections that Bush desired to see. However, the real body of knowledge is not made of mechanical links. It consists of authors like Oskin and coauthors, the editors who review articles (including editors who misunderstand the ideas), all the readers who found something useful in these articles.

REFERENCES

1. V. Bush, "As we may think," *Atl. Mon.*, vol. 176, no. 1, p. 101, July 1945.
2. D. Deutsch, "Quantum theory, the Church-Turing principle, and the universal quantum computer," *Proc. R. Soc. A Math. Phys. Eng. Sci.*, vol. 400, no. 1818, pp. 97–117, July 1985. doi: 10.1098/rspa.1985.0070.
3. P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. 35th Annu. Symp. Foundations Computer Science*, Santa Fe, NM, 1994, p. 124. doi: 10.1109/SFCS.1994.365700.
4. M. Oskin, F. T. Chong, and I. L. Chuang, "A practical architecture for reliable quantum computers," *Computer*, vol. 33, no 1, pp. 79–87, Jan. 2002. doi: 10.1109/2.976922.
5. M. Fleischmann and S. Pons, "Electrochemically induced nuclear fusion of deuterium," *J. Electroanal. Chem.*, vol. 261, no. 2, pp. 301–308, Apr. 1989. doi: 10.1016/0022-0728(89)80006-3.

DAVID ALAN GRIER is a principal with Djaghe, LLC. He is a Fellow of the IEEE. Contact him at grier@gwu.edu.