“The History of Computing: An Introduction for the Computer Scientist”

by

Thomas Haigh
www.tomandmaria.com/tom

This article is excerpted from Atsushi Akera & William Aspray (eds.), *Using History To Teach Computer Science and Related Disciplines*, Computing Research Association, 2004. It constitutes pages 5 to 26 of that volume.

The article is complemented by an on-line document
“Key Resources in the History of Computing”, available at
http://www.tomandmaria.com/tom/Resources/ResourceFile.htm

2. The History of Computing: An Introduction for the Computer Scientist

Thomas Haigh
University of Wisconsin–Milwaukee

Abstract
This paper is intended to provide an introduction to the study of the history of computing for someone who has a good knowledge of computer science, but only a limited knowledge of the history of computing and no formal training in any kind of historical study. It has three parts. The first outlines the development and current state of the history of computing as an area of academic study. The second examines history as an academic field of study, outlining its research methodologies, the various historical subfields relevant to computing, and the ways in which historical research differs from computer science research in its goals, assumptions, and practicalities. This section also specifies in general terms which areas of the history of computing have been studied so far and which ones are currently languishing in obscurity. The final section offers some personal thoughts on different ways in which attention to the history of computing might help to make a computer scientist into a better teacher and, on a broader level, might make computer science itself into a better discipline.

This paper should be read in conjunction with my detailed description of "Key Resources in the History of Computing" (see chapter 26). Where that guide provides an annotated list of specific online, published, and institutional resources in the field, this paper is more concerned with giving the reader a sense of what it is that historians of computing do, why they do it, and how they are different from computer scientists.

The paper has three parts. The first outlines the development and current state of the history of computing as an area of academic study. The second examines history as an academic field of study, outlining its research methodologies, the various historical subfields relevant to computing, and the ways in which historical research differs from computer science research in its goals, assumptions, and practicalities. This section also specifies in general terms which areas of the history of computing have been studied so far and which are currently languishing in obscurity The final section offers some personal thoughts on different ways in which attention to the history of computing might help to make a computer scientist into a better teacher and, on a broader level, might make computer science itself into a better discipline.

The History of Computing

Earliest Developments

The study of the history of computing is at least as old as the electronic computer. The creation of the first experimental programmable electronic computers in the late 1940s, followed by the first commercially produced models in the early 1950s, created a market for books and articles explaining these electronic marvels to a bemused public. The authors of these pieces wasted few superlatives in celebrating the unprecedented speed and power of these machines. Indeed, the earliest and most influential of the books was entitled *Giant Brains, or Machines That Think*, a title that more sober computer experts spent decades trying to dispel from the public imagination.

Wide-eyed awe, however, only got the popularizers so far. In trying to explain what the new computers did and how they did it, the early computer promoters and salesmen quickly turned to homely analogies and comparisons with earlier and more familiar technologies. Because the first computers were designed and used to perform scientific calculations, Edmund

C. Berkeley, author of *Giant Brains*, presented the computer as the latest in a long and unbroken line of calculating aids that reached back through the slide rule and desk calculator to the abacus. This invocation of Pascal's calculator and Napier's bones became a familiar part of introductory guides to computing, presented in television series, children's books, and public exhibitions. With the success of electronic computers came minor celebrity for nineteenth century British inventor Charles Babbage, whose unsuccessful attempts to construct a programmable computer had been forgotten long before his death in 1871.[3]

Until the 1970s, the history of computing was largely confined to this role as the source of material for the introductory pages of an article or chapters of a book, designed to make the unfamiliar less threatening. But as the pioneering generation of academic and commercial computer experts began to sense their own mortality, some of them developed a new concern with documenting and preserving the early history of electronic computing, such as the ENIAC project, the first commercial systems produced by IBM and UNIVAC, and the work of computing teams at places like the RAND Corporation and the Institute for Advanced Study.

The first major organized project in the history of computing was an extensive series of oral history interviews in the early 1970s. The project was sponsored by AFIPS, the now-defunct consortium of computer-related academic societies, and was carried out in collaboration with the Smithsonian Institution. These conversations with many of the most important figures in computing from the 1950s were recorded and transcribed for posterity. Following the pattern of most early work in the area, the project leaders were concerned primarily with hardware, and paid little attention to software and still less to the usage of computer systems.[4] Around the same time, some participants began to turn their attention to memoir-writing, and in some cases became keen amateur historians of their own field. Among the most prominent examples was Herman Goldstine, a collaborator of John von Neumann in the very early days of electronic computing, who wrote an important book called *The Computer from Pascal to Von Neumann,* which blended the earlier history of computing with his own perspectives on events of the 1940s.

Three Key Institutions for Computing History

The three most important institutions in the history of computing all date from the late 1970s. The first of these is the journal *Annals of the History of Computing*[5] (now *IEEE Annals of the History of Computing*). Originally published by AFIPS, *Annals* (as it is informally known) has developed along with the community it serves. Its first editor-in-chief, Bernie Galler, and almost all the active members of its original editorial board were distinguished computer scientists. These included software engineer Brian Randell and COBOL designer Jean Sammett. The journal printed a wide range of material, running memoirs and anecdotes alongside scholarly articles, printing the transcripts of reunions, devoting special issues to historically significant machines such as the IBM 650, and even featuring a regular section entitled "Meetings in Retrospect." Its rare controversies were usually personal in nature, the most heated involving questions related to the invention of the computer.

[3] Berkeley, Edmund C., *Giant Brains or Machines That Think* (New York, 1949).
[4] These oral history collections are still available for reference in the Archives Center of the National Museum of American History http://americanhistory.si.edu/archives/ac-i.htm, though at the time of writing almost none exist in electronic form.
[5] http://www.computer.org/annals.

While *Annals* today still contains many of these elements, it has gradually evolved into something more akin to a conventional academic journal. The demographic balance of its editorial board has gradually shifted away from eminent computer scientists and toward a younger group trained as historians. Memoirs, while still common, are increasingly being joined by more scholarly and analytical peer-reviewed articles. There has been a corresponding, and far from complete, shift away from articles concerned primarily with documenting stories that would otherwise be lost and toward those attempting to place events in a somewhat broader social or intellectual context.

The second major institution in the history of computing is the Charles Babbage Institute.[6] The Babbage Institute found a permanent home at the University of Minnesota campus in Minneapolis, thanks to the support of the university and of the once-thriving local computer industry. Though it has remained quite small (with a full-time staff of about half a dozen), CBI provides vital services to the historical community. Its primary role is as an archival repository. On a physical level, archives typically consist of a large, climate-controlled storage area in which boxes full of papers, photographs, and other historical materials are housed. Researchers request particular boxes and consult them in a small reading area. The archive is responsible for acquiring, sorting, and cataloging the papers, which are generally donated from the internal files of firms and other organizations or the personal collections of individuals. Historians generally use archives as the main source of material for their research. A steady trickle of acquisitions has given CBI an unrivalled collection of computer-related materials, from large collections such as the records of computer manufacturers Burroughs and CDC and of the Data Processing Management Association, to smaller collections such as the personal correspondence of Daniel McCracken, an early computer textbook author and ACM President.

CBI also houses the largest collection of computer-related oral history interviews, almost all of which have been transcribed and abstracted. The full text of most of these interviews is available online, including such luminaries as Donald Knuth, Marvin Minsky, Gene Amdahl, Thomas J. Watson, Jr., J. Presper Eckert, and Edsger W. Dijkstra. Over the years, CBI and its staff have received a number of research grants, including one to examine the role of ARPA in fostering the development of computer science, and another to explore the history of software. A reprint series, now dormant, brought important but hard-to-find historical documents, including papers on computing by Babbage, Alan Turing, and John von Neumann, back into print. Finally, CBI administers the Tomash Fellowship, which supports one graduate student working on a dissertation topic in the history of computing each year.

The last of the three major institutions in the history of computing is the Computer History Museum.[7] This holds an exceptional collection of rare and historical computer hardware, including pieces of the ENIAC, an Enigma machine, a SAGE console, a Cray 1 supercomputer, a Xerox Alto, an Altair, and an Apple 1. It also boasts large collections of photographs and software. Today, the museum is approaching its first anniversary in an attractive and spacious new home in Mountain View, California (part of Silicon Valley). Its exhibits are slowly taking shape, although a large amount of hardware is already on view with a minimum of explanation. The museum runs a high-profile speaker series and awards program, publishes a glossy newsletter, and maintains an attractive website. Like *Annals* and CBI, its origins can be traced back to the 1970s, though via a more circuitous path. It is a descendant of the now-defunct Computer Museum, once a successful attraction in downtown Boston. This earlier incarnation was closely tied to DEC, having begun life as the Digital Computer Museum in the lobby of a

[6] http://www.cbi.umn.edu.
[7] http://www.computerhistory.org/.

DEC building. By the 1990s, the leading local firms on which it relied for sponsorship were in terminal decline, and the museum closed. Much of its collection, however, ended up in storage in California, and became the nucleus of the new museum. The new museum is still taking shape, and so far at least has understandably focused primarily on fundraising and work with the volunteer community of computer enthusiasts, rather than on the support of historical research or the management of archival resources. It may evolve to become the primary interface between the history of computing community and the general public.

These are not the only institutions, of course. The Smithsonian includes several curators with an interest in the history of computing, and for the past decade has had an impressive array of hardware on display in its "Information Age" exhibit. It also holds several important collections of computing-related records in its archives. In recent years, a small group called the Software History Center[8] has been active in organizing oral history events related to the history of the business software industry. In Britain, the National Archive for the History of Computing[9] oversees records related to the history of British work in this field, while the Heinz-Nixdorf Museum[10] in Paderborn, Germany displays an excellent collection in an extremely stylish setting. Most of the world's major museums of science and technology, including the Science Museum[11] in London and the Deutsches Museum[12] in Munich, provide a reasonably extensive display of historic computers. There are also amateur groups scattered around that focus on the preservation and restoration of old hardware, and a number of smaller museums built around personal collections.

Since the emergence of its key institutions in the late 1970s, the history of computing community has gradually expanded. It has a reasonably coherent core, closely tied to *Annals*, CBI, and the Computer History Museum. This core group is quite small and quite chummy, in that we know each other well and tend to encounter each other once or twice a year at the *Annals* board meeting, related conferences, or special workshops or meetings sponsored by one group or another. If one combined the editorial board of *Annals* (which includes specialists from numerous universities in the United States and Europe, the Smithsonian, and CBI) with the staff of the other two institutions then one would have identified most of those who earn their living in the field, or focus their intellectual energies on it. There is, of course, a much larger group of people with a secondary interest in the field, including journalists and other writers who might be working on a book or article related to the history of a particular firm or individual, and numerous older people who lived through the history concerned and might contribute articles or personal memoirs on fields of particular interest.

Historians and the History of Computing

Professionals and Amateurs

The core group studying the history of computing is a mixture of eminent computer scientists and other old-time computing people with what one might be tempted to call professional historians (a tricky term, given that history is not a profession). However, "professionally trained historian" is a better defined category, and could reasonably be applied to anyone with an M.A. or Ph.D. degree from a reputable program in history or a specialized

[8] http://www.softwarehistory.org/.
[9] http://www.chstm.man.ac.uk/nahc/.
[10] http://www.hnf.de/.
[11] http://www.sciencemuseum.org.uk/.
[12] http://www.deutsches-museum.de/e_index.htm.

history of science, technology, or business program. Most professionally trained historians active in the history of computing hold university positions (some in history programs, some in various "science, technology and society" programs, and some in information science schools), or work in museums as curators. A handful work as historical consultants, and a few more do historical work as a hobby while holding down more lucrative and unrelated day jobs with no historical content. There is also a shifting crowd of journalists, who usually wander into the field for a book or two and then wander off again to write about something different.

Over time, the balance between these groups has shifted toward the professionally trained historians, and this trend appears likely to continue in the future. The split is not absolute because many of the professionally trained historians studied or worked in computing before being trained in history. (This perverse career move at least provides them with one good joke to start their presentations with.) Professionally trained historians tend to take different approaches to their work from those favored by journalists or computing practitioners. They would tend to believe, fairly or not, that journalists are primarily interested in selling books by turning every story into a variation on the "X the lone genius who invented the Y and overcame decades of scoffing to revolutionize Z." They would also complain that amateurs with a background in computing tend to be very good at amassing facts and details, particularly about computer hardware or events that they personally took part in, but aren't very interested in addressing "broader historical questions" (by which they mean the kinds of things that historians talk to each other about).

Professionally trained historians, in contrast, tend to write in order to impress other professionally trained historians. You can't blame them for this, since every academic discipline, including computer science, takes the same inward turn as an essential part of the development of a new academic field.[13] One wants to publish work on hot topics, using new and exciting approaches. As a result, one gets hired by a good school and invited to the best conferences and workshops, and sees one's articles cited frequently. Funding priorities also play an important role in computer science, and obviously less so in history. Combined with the immersion of graduate students within specific cultures, this steers individuals in particular directions and, on a broader level, shapes the field as a whole.

Indeed, historians feel a lot more guilty about this than most scientists, since they don't use any equations; thus, they are haunted by the nagging sense that if they could just improve their turgid prose styles then maybe the public would be snapping up their books rather than the work of those journalists who craft huge presidential biographies, histories of the Civil War, and the recent spate of books about fishes, spice, or Jesus. This almost never happens, although with care an historian might reach a specialized non-academic audience of buffs.

How History Differs from Computer Science

The motivations and practices of academic historians have a lot in common with those you are familiar with in academic computer science. Both spend years in graduate school taking specialized courses and comprehensive exams before writing a dissertation and receiving a Ph.D. Both usually aspire to a tenure-track job at a good university, and pursue this goal by doing novel research, publishing in prestigious peer-reviewed journals, flattering their senior

[13] In the early days of computer science, for example, theory and mathematical rigor were sought above all else by elite departments such as Stanford, in order to set aside computer science from mathematics on the one hand and service-oriented computer center work on the other.

colleagues, and accumulating passable teaching evaluations. Both trade material success for intellectual freedom.

Closer up, however, differences appear. Compared with typical computer scientists in the same institution, historians usually earn less money, teach more courses, and graduated from better schools with stronger resumes. This is a function of market forces because there are many more qualified historians for each advertised job, and because the only other significant source of professional employment for Ph.D. historians is museum work. Museum work doesn't exactly compete with Microsoft and Intel in pumping up salaries for top researchers.

There are no industrial or government research labs in history; in contrast, computer science has institutions like IBM Research, Bell Labs, Xerox PARC, Argonne, and Los Alamos. The reason is that the social motivations behind research in the humanities are fundamentally different from those in hard science and medicine. Society pays for research in medicine because it wants to cure the sick; at a lower level, it pays for research in computer science because of the clear benefits of its existing products such as the Internet, and because it might be helpful in finding new ways for the armed forces to more efficiently kill just the right people at minimum risk to themselves.[14]

On the other hand, society's main interest in research in the humanities is as a way of deciding which professors deserve to be hired or tenured and of encouraging them to stick around and teach more undergraduates. This makes more sense than you might think because an undergraduate degree in history, unlike a graduate one, is a very sensible major for an ambitious young person. It teaches students to read sources critically, assimilate large volumes of material of different kinds, write persuasive and factually based arguments, and reason with incomplete information. This is excellent preparation for law school, for Congress (Newt Gingrich claimed to be a history professor), and for even higher office. America's current leader has a degree in history from Yale, though I think it is fair to say the discipline isn't eager to take credit for his skills in the creative reading of intelligence sources.

Important historical research is carried out very differently from important research in computer science. A major research project in computer science is likely to receive hundreds of thousands of dollars in funding, employ several post-doctoral students, fill a lab or two with expensive equipment, provide dissertation topics for half a dozen Ph.D. students, and produce a long series of technical reports and papers, each with between three and six authors. It might well involve collaboration with other institutions or with industrial research labs.

A major historical research project, on the other hand, would typically be done by one person working for perhaps eight years. The result might be two long, painstakingly refined papers in good journals and one book of about 300 pages published by a leading university press. The major unit of scholarly production and exchange in history is the single-authored book, not the journal article, and certainly not the conference paper.[15] Many of the most

[14] The description of the social function of research in different fields is paraphrased from Robert X Cringley, *Accidental Empires* (New York, 1996). Historians footnote their borrowings religiously because failing to cite could be a career-ending mistake. This isn't to say that computer scientists are pawns of the military, and indeed they've proved very good at getting military money and then doing whatever it was they wanted to do anyway.

[15] Few historical conferences even publish proceedings, and most presentations count for about as much as a book review when judging a vitae. The papers themselves are not usually reviewed prior to acceptance, though abstracts and resumes are. That is one of the most important differences between

important books in history began as Ph.D. dissertations, refined and improved for a few years after landing a first academic job. One book would usually be enough for tenure at a decent university, with two the benchmark for elite research-oriented institutions.[16] Research grants are comparatively rare in history, and are usually for no more than a few tens of thousands of dollars—for example, to pay for a trip to an archive and a graduate research assistant for one semester. They are most likely be received by historians working in fields like the history of medicine or business where well-endowed foundations have an interest in the field. There is no real equivalent for the role played by NSF and DARPA in computer science. The lack of research funding makes postdoctoral fellowships very rare in history, so even a successful young historian is likely to spend several years in visiting appointments with heavy teaching loads before landing a tenure-track job.

Historical research is thus a solitary kind of affair with little collaboration.[17] One positive result of the lack of research funding is that, from the thesis on, the researcher has enormous flexibility in picking a topic to work on and in shaping the direction of the project. A student is not at the mercy of her advisor in being given a topic and hoping to receive reasonable billing on some of the group's papers. On the other hand, this leads to a much more detached relationship with less of an investment in the student's work. Historians, ironically, never bother to prepare elaborate academic genealogies of the kind popular in theoretical computer science.

History job announcements rarely specify an interest in the history of computing. This means that a hopeful researcher in the area instead has to market herself as a special case of an historian of something else, who happens to be interested in computer-related topics. The problem is that the people doing the hiring will almost certainly know nothing about the history of computing, and will probably see it as peripheral to the researcher's main area of interest. I am not personally aware of any case in United States in which an individual has written a dissertation focused primarily on any aspect of the history of computing, and then been hired by a different institution for a tenure-track job including research and at least some teaching in the history of computing.[18] That surely must be the minimum standard by which to claim that an area has begun to function as an accepted disciplinary specialization.

---

history and computer science (or indeed science in general), where books are usually textbooks and count for very little when it comes to tenure or hiring decisions.

[16] I should also make clear that, unlike textbooks in computer science, these history books are no more likely to make appreciable sums of money for their authors than are journal articles. They are purchased primarily by libraries, and a book selling several thousand copies would be considered an enormous success. Advances would not usually cover the Xeroxing expenses, let alone the author's time. The books are effectively subsidized by the university employing the author or, given the teaching loads common in the humanities, are produced in the author's evenings and weekends. So here, too, "pull" from possible purchasers counts for a lot less than the "push" provided by tenure committees. The fact that the public has an insatiable appetite for books about Nazis, presidents, and wars has therefore not made these topics particularly attractive to professional historians. (This is analogous to the disregard shown by computer science programs to the stated priorities of industrial recruiters.)

[17] Jointly authored books and papers are by no means unknown in history. This point was brought up in discussion at one of the workshops. In response, William Aspray, who has worked on numerous joint projects, pointed out that while this can be a way of getting additional perspectives into a piece of research, it might actually prove to be more work to write an historical piece jointly than to handle it individually. This is because history is so tied up with writing, which is hard to divide up into self-contained activities.

[18] Things are not quite as bad as this suggests. There are knowledgeable people currently teaching and publishing on the history of computing in the United States, but they were hired to do other things or stayed on at their Ph.D.-granting institution. In other cases, people with specialist training in the history of computing found jobs in other fields (such as information science, science and technology studies, and

As peripheral as the position of historians of computing is in historical fields, they have proved to be much more welcoming than computer science departments. In professional fields such as medicine, library and information science, and even business, a handful of leading schools have hired historians and added history courses to their core curriculum. This has never happened in any American department of computer science, and there are no signs that it is likely to happen any time soon.

Different Kinds of History

So far I've talked about history as if it was a single field. In fact, history is a highly segmented academic discipline. Many job announcements in computer science state several general research areas of interest, and hint that exceptional candidates in other areas will be considered. In contrast, history jobs usually stipulate a geographical region, a time period, and a methodological approach. A school might seek a historian focused on the United States in the second half of the nineteenth century, working on issues of business, labor, and social history. Or it might emphasize Western Europe during the Renaissance, looking at the history of the physical sciences. One recent opening even specified expertise in the history of Delaware, among a long list of other requirements. The job market is such that search committees rarely have to compromise.[19]

The point of telling you all this is not to fill you with sympathy for the plight of the historian, but rather to give you a sense of why the history of computing has developed so differently from research areas within computer science, where the emergence of SIGs, journals, influential conferences, and influential centers of research has often been comparatively smooth. Because research funding is scarce and collaborative research unusual, research progresses fitfully and according to the whims and interests of the individual researcher. There are no tenured, well-funded professors with armies of graduate students ready to fan out and probe different aspects of the field. Although there have been a few attempts to convene the great and good to identify future research agendas for this field, these have been hindered by the lack of researchers to direct and by the fact that the history of computing community cannot collectively hire, fund, or tenure researchers. So research priorities rest not just on the whims of individual professors and graduate students, but on the whims of individual professors and graduate students who have to meet the standards of a wide range of different fields.

Some of these weaknesses are also strengths. One positive result of the proliferation of historical specialties, coupled with the non-existence of the history of computing as a recognized academic specialty, is that those who do choose to work on topics in this area bring with them the techniques and concerns of many different historical fields. As historians gradually come to grips with the events of the past half-century, many more historians will surely recognize the

---

informatics) in which no historical teaching was possible, but have continued to research and publish on historical topics.

[19] Overall, one of the most important factors shaping research directions has been the kinds of people hired to academic jobs, which in turn has been shaped by the kinds of courses taught to undergraduates. In the 1960s and 1970s, expansion in university education and changing social attitudes sparked a turn toward social history, and an examination of representative experiences rather than studies of elites. Examination of class, race, and gender became the central concern of historians. More recently, this has been supplemented by attention to cultural history, and the study of things like popular films and magazines. Much hiring is taking place in identity-fields such as black history, Jewish history, gay and lesbian history, and so on.

importance of computer technology to crucial developments in science, business, culture, and daily life.[20]

Here are some of the main historical subdisciplines with clear conceptual connections to the history of computing:

***History of Science**:* History of science is quite well established as an academic field. There are a number of free-standing history of science departments in leading American universities such as Harvard and the University of California, San Diego, and there is a tendency to include a lone historian of science in many of the larger departments of history. It runs from pre-history onwards, with some scholars focusing on classical science and many on the scientific revolution and the enlightenment. Historians of science usually specialize in geographic areas and historical time periods, such as ancient China or twentieth-century Latin America. Many interdisciplinary programs in "science studies" or "science, technology and society" at universities such as Cornell include strong historical dimensions. The major society in this area is the History of Science Society, whose journal, *ISIS*, is the leading American publication of its kind.

Computer technology plays an increasingly vital set of roles in almost all scientific disciplines. Examples include data reduction and experimental control in physics, modeling efforts in meteorology and economics, gene sequence reconstruction, statistical analysis, scholarly collaboration, and information dissemination and retrieval. Computers have changed the laboratory practices, publication patterns, analytical methods, and standards of evidence and proof in many fields. In some areas they have provided entirely new avenues for experimental work. As early as the 1960s, computing was radically reshaping disciplines such as numerical analysis. Few of these topics have received any attention yet from historians of science, but this will surely change. In addition to scientific computing, the other big topic for historians of science should be the history of computer science itself. Computer science, and the related fields of information science and software engineering, were among the most important disciplines to emerge in the past half-century, and have begun to attract some historical work (so far primarily by members of the disciplines concerned).[21]

History of Technology: As a field, the history of technology is a little smaller and less well established than the history of science, but may be catching up. In America, its roots lie primarily in the history of engineering (largely practiced, for obvious reasons, in schools with strong engineering programs) and in the efforts of curators at the Smithsonian and other museums. The Society for the History of Technology has almost two thousand individual members, and publishes *Technology of Culture*, the field's leading journal. Historians of technology are paying increasing attention to the use and consumption of technology as well as its design and production, and are expanding the range of technologies they study beyond large, metal things such as railways and space rockets and toward things like zippers, hair products, and personal stereos. Much work in recent decades has been framed as an examination of the processes by which technologies are shaped by social processes and

[20] As a rule of thumb, it takes about fifty years after the events concerned for historical scholarship on a particular area to really mature to the point at which historians have come to understand the events of a particular era in ways different from those held by the people who lived through it. This achievement of "historical distance" takes place partly through the death of those concerned, who tend to have strong and partisan notions of what happened.

[21] See the sources cited in "Key Resources in the History of Computing" (chapter 26, this volume), under the heading *Computer Science*.

interests, but in turn reshape society through their use. Historians of technology have so far paid rather more attention to computing than have historians of science. Most work thus far has addressed mainframes, minicomputers, and supercomputers, and has paid particular attention to the evolution of the computer from earlier technologies and to the roles of government and business organizations in steering that evolution. However, it seems certain that, for the 1970s onward, the proliferation of personal and embedded systems in areas such as cars, home systems, PDAs, cellular phones, navigation systems, domestic appliances, audio and video devices, and Internet communications will move computer-related topics toward the heart of the history of technology.

***Business History**:* Business historians are found in both history departments and business schools. The subject is fairly marginal in both kinds of school, with a few notable exceptions such as the Harvard Business School. Traditionally, much work in business history has consisted of celebratory corporate histories commissioned to mark major anniversaries. Successful or prominent firms, such as Microsoft, often attract hastily written books by journalists looking to cash in on the huge market for easy-to-read business stories. However, academic historians often produce more analytical and critical work, and have been paying increasing attention to the evolution of business institutions (such as the multinational corporation), the development of managerial specialties, the creation of new products and industries, and the relationship between corporate structures, markets, and customers. The leading American society in this area is the Business History Conference, which now publishes a journal called *Enterprise and Society*. This competes with Harvard's longer-established *Business History Review*. A considerable amount of work has been published on the business history of the computer industry, including a shelfful of books on IBM and studies of the other early computer companies. Professional historians have done relatively little work on the prominent firms of the microcomputer and Internet era, in part because none have made their archives available.[22]

Separate from the business history of computer firms, and so far much less well developed, is a literature on the use of computer technology in different kinds of organizations. Computers have long played a vital role in business administration. More recently, firms have redesigned their core processes and activities in new ways made possible by computer technology. In addition, computers may finally be living up to their promise of reshaping organizational structures, eliminating many kinds of middle management positions, and supporting managerial decision making through the integration of knowledge and data from different parts of the firm. Corporations rely on a huge new class of expert technicians to keep their operations running. While few business historians have yet written about these changes, they will be very hard to ignore.[23]

Economic History: Economic history is not entirely distinct from business history, though the division has been growing wider in recent years. Economic historians tend to be trained as economists, and apply modern quantitative methods to data sets gathered from historical

[22] Pretty much every computer firm of any note has attracted at least one history, most of them written by journalists (around the peak of the company's fortunes) or participants (often long after the firm in question vanished). Professionally trained historians have written a number of business histories, notable among which is Martin Campbell-Kelly, *ICL: A Technical and Business History* (New York, 1989).

[23] A number of works on the use of computers in business by Thomas Haigh, JoAnne Yates, and James Cortada are cited in the resource guide. Another example, produced by a Harvard Business School team, is James L. McKenney, Duncan C. Copeland, and Richard O. Mason, *Waves of Change: Business Evolution through Information Technology* (Boston, MA, 1995).

sources. The main organization for economic historians is the Economic History Association. Much economic history concerns fairly technical topics such as the comparative history of central banking in different countries, or trade patterns in the classical world. Economists are sometimes attracted to historical examples as a way of testing broader theories, which often leads to repeated reexamination of particular topics from competing viewpoints while other topics go unexplored. For example, economists have looked at earlier waves of technological investment in railroads and electrical power to compare their magnitude, returns, and duration with the recent wave of spending on computer technology. They have also argued about whether the continuing dominance of the QWERTY keyboard layout (famously created to slow typists down) proves that historical accidents can lock society into suboptimal technologies (a process known as "path dependence.") Other computer-related topics include the management of high technology innovation, and the potential role of government in protecting and encouraging nascent technological industries.[24]

Other Areas: Many other historical specialties can, or should, interpret aspects of the history of computing according to their own interests. For example, a few decades ago a number of prominent labor historians and sociologists developed an interest in the role of computerized automation in transforming skilled and manual work in areas such as factory work, banking, and clerical labor.[25] Oddly, research in this area has declined sharply since the 1980s, but an eventual resurgence seems likely. Many historians study the role of government and its many agencies, departments, and programs, but so far have paid little attention to the role of computers and punched card machines in shaping its development.[26] However, quite a lot has been written about the role of government funding in the development of computer technologies.

How might these fragmentary perspectives prove to be an asset? Consider, for example, the history of operating systems. Different groups might see it in very different ways. An historian of science might focus on the computer science side of the story, looking at Dijkstra and his conceptual work on semaphores, or use the commercialization of time-sharing as an example of transfer of skills and ideas from academic researchers to industrial groups. An historian of technology might be more interested in the role of System/360 and its much publicized problems in the development of software engineering as an academic field. A business historian, on the other hand, would be much more interested in the history of Microsoft or Novell as corporations, looking at marketing techniques, technological strategies, management methods, and so on. Or, focusing more on applications, the business historian might decide to examine the role of operating system developments in facilitating the implementation of real-time, database-oriented applications for management and control purposes. A labor historian, in contrast, might see something completely different. The term "operating system" was coined because it was supposed to replace the work of computer operators (themselves often retrained punched card machine operators). Over decades, it shifted a large amount of work between operators and application programmers, and created a new specialty of systems programmers. An historian interested in the history of government

[24] One example of an economic history approach to a topic in the history of computing is David Mowery, *The International Computer Software Industry: A Comparative Study of Industry Evolution and Structure* (New York, NY, 1995).

[25] Among the most notable of these were Harley Shaiken, *Work Transformed: Automation and Labor in the Computer Age* (Lexington, MA, 1986); and Shoshana Zuboff, *In The Age of the Smart Machine:The Future of Work and Power* (New York, NY, 1988).

[26] A notable work addressing the history of computing and other information technologies in the development of the British government has recently appeared, and may spark some interest in the topic. Jon Agar, *The Government Machine* (Cambridge, MA, 2003).

support for research and development might focus on the role of military projects in supporting developments, including the SAGE project of the 1950s and ARPA funding of time-sharing.

Historical Research Methodologies

As diverse as these different kinds of history are, there are still some fairly safe generalizations that can be made across them. For the computer scientist, the most important one to grasp is that historians do not see themselves as practicing a science, even a social science. This has profound effects on the structure and practice of historical research. Social sciences such as sociology, economics, and specialist fields such as information science have become increasingly mathematical and quantitative in a quest for scientific rigor. Researchers in these areas win acclaim by gathering precisely controlled data sets, framing and testing extremely narrow research questions, and achieving results with a high level of statistical significance. Their research papers include long and explicit discussion of methodology, and segregate descriptive presentations of results from analysis and analysis from conclusions. They embrace the turgid language, passive voice, and specialist vocabularies beloved of the harder sciences.

Historians, in contrast, see themselves as practicing a craft. Their job is to take the mass of written documents and other sources (such as artifacts and audio recordings) preserved from the past and use them to tell stories or, as they would put it, to craft historical narratives. Social or hard scientists sometimes pick up the work of historians and frown in bafflement at its apparent lack of rigor. After a brief introduction, an historical work usually tells a story, generally in historical order. In a longer work, each chapter tells a self-contained part of the larger story. There is no lengthy framing of research questions, no detailed defense of data-gathering and analytical methodology, and no long presentation of conclusions and recommendations. No historian bothers to ritually cite seminal papers by the founding fathers of the discipline or exhaustively survey all previous work in the field. The result, our hypothetical scientists might feel, is just a descriptive presentation of raw data, lacking entirely in theory, analysis, or intellectual framework. History work would, in the famous phrase, consist of nothing more than writing down one damn thing after another until the book was full.

That would be a fundamental misreading of the historian's craft. During their graduate training, historians spend a great deal of time learning research methods, critiquing the work of others, discussing the strengths and weaknesses of different approaches, and practicing different techniques. Any good historian puts an enormous amount of thought into framing a coherent and meaningful research structure, defining historically accurate and analytically useful categories, and tying the specific story being told into broader issues, such as the development of the middle class or the relationship between race and gender. The more fundamental difference is that social scientists make a ritual and energetic display of all this work in the final product, whereas historians are trained to hide the evidence of their labor to leave the final narrative as unencumbered as possible. I like to think of this as being akin to the work of a sculptor who uses many kinds of molds, scaffolds, and sketches during his labors, but disassembles and destroys these as the work progresses, creating a final product that gives an impression of effortless verisimilitude to life. The social sciences, on the other hand, are more like postmodern buildings, such as the Pompidou Center in Paris, which display their internal workings on the outside so as to draw attention to them.

Historical research is inseparable from the telling of a particular story. As Michael Mahoney noted during discussion of this point at the conference, a computer scientist who says that the research is finished and just needs to be written up is indeed nearing the end of a

project. An historian who says the same thing has not really started. Indeed, one of the most important pieces of advice given to graduate students in the field is to write up material from the very beginning to see how the story hangs together and what additional material is needed to tell it. Without doing this, one might accumulate thousands of pages of notes that will never coalesce into a coherent project. Because of this focus on narrative, historians tend to be better writers and tend to take teaching more seriously than do computer scientists. There are, of course, many exceptions to this generalization on both sides.

Theory is not a prestigious area within history. Historians do sometimes produce articles urging more attention to some particular approach or topic, and journals sometimes publish articles reviewing existing work in an area, categorizing its strengths and weaknesses and setting out a possible agenda for future work. They call this "historiography." In general, however, the editors and reviewers of historical manuscripts like to keep such discussion to the minimum necessary to show some kind of relevance to earlier work and imply that a novel and important contribution of some kind is being made. Historians are deeply suspicious of anything more abstract, and are usually amused more than threatened by occasional efforts to produce grand unifying theories of historical change, or to apply cybernetics or mathematics to the discipline.[27]

Historians decry "presentism"—the tendency to interpret historical events and ideas in terms of our present-day interests and assumptions, rather than doing the work necessary to understand what they meant to the people involved. For example, the team responsible for the database query language SQL originally believed that they were producing a tool for non-technical managers to directly query databases using an easy-to-grasp subset of English. Only much later did they realize that its main use was as a lingua franca for application programs to communicate with databases and for the transfer of data between databases. It did not solve the problem it was created to deal with, but thrived anyway as the solution to a different one. One can't understand why SQL took the form it did without appreciating this.

A related practice is "Whig History," which is a story consisting entirely of deliberate and unbroken progress from primitive beginnings to current perfection. In this view, the only important things in history are the "contributions" made at various points to our current understanding of the world. The history of computing is especially vulnerable to these problems because the real, sustained, and rapid progress made in many areas of hardware and software development does tempt one to view the past as a simple march of progress. One problem with this view is that a history consisting only of progress towards a fixed goal has erased the most valuable lessons we might learn from it on topics such as the need for flexibility, the difficulty of foreseeing the future, ideas that were bad then and shouldn't be tried again now, and even ideas that were bad then but might work now.

While rejecting scientific methodologies for their work, historians do display a profound, almost obsessive, attachment to certain kinds of rigor. The methodology banished from the text is displayed instead in the footnotes, which might (at least before editors go to work) comprise a full one-third of the total length of manuscript. While explicit discussion of methodology does sometimes occur here, it is more important for them simply to list the source of every quotation

[27] A generation ago, in the late 1970s and early 1980s, there was a push toward extensive use of statistics and databases in history, but these approaches have almost vanished from mainstream historical training. They survive only in interdisciplinary areas such as historical demography and economic history, but these fields have been left almost entirely to those trained as demographers and economists.

or fact presented by the author. The skilled reader is able to essentially reverse-engineer the author's methodology from these citations. Indeed, faced with a new work in his own field the historian frequently turns first to the footnotes, scanning them to determine the range of sources, quality of archives, number of different collections, geographical range, social diversity, and general heft of the evidence upon which the author has drawn. A paper about a famous part of computer history, say the development and use of VisiCalc (the first spreadsheet) based largely on existing popular histories and reading of a single newspaper, will impress much less than one based on corporate records, hitherto unknown personal correspondence, laboriously cross-referenced consultation of computer dealer trade journals, user group newsletters, popular computing magazines, and interviews with surviving users, developers, and dealers.[28]

This stress on using a diverse range of sources to present a balanced picture means that historians rarely espouse a doctrinaire commitment to one or another theoretical framework. They are often pragmatists at heart. Most historians eschew the flamboyant postmodernism popular in departments of literature. Thoughtful historians have always known that there can be no neutral, complete, and empirical description of the past, and that any historical narrative must make assumptions, stress some things over others, and even favor the perspectives of one group over another. Each generation inevitably reinterprets the past in the light of its current preoccupations, which is why medieval and classical historians still have useful work to do. Yet historians do not conclude from this that it is unnecessary to strive for honesty and fairness in their craft, or that it is futile to pile footnote upon footnote, or that all possible accounts are equally valid. Indeed, sins such as the failure to properly cite quoted half-sentences, or an apparently deliberate misrepresentation of archival evidence, have led to the disgrace of more than one prominent historian in recent years. It may be that historians are too extreme in their erasure of methodology, too skeptical of big ideas, and too stodgy in their reliance on footnotes, but theirs is, by and large, a noble and unpretentious calling.

Topics Covered and Not Covered

Given the smallness and diversity of the history of computing community, it is not surprising that its coverage of important historical topics has been far from uniform. Scholars working on some historical topics, such as the Civil War (British or American) or the politics of the New Deal, must do a lot of work to identify a topic not already written about, or to show that their insights render inadequate a shelf of existing books on the topic. In the history of computing, however, almost no topics are over-studied. The only three exceptions that spring to mind are journalistic profiles of Bill Gates, arguments about who really invented the computer, and discussions of the ENIAC.

Several attempts have been made to write overall histories of computing. Their scope has broadened over time, from histories of calculating devices (a topic that historians came to understand quite well) to more ambitious attempts to present the computer as an evolution of existing technology in fields such as business administration. This, of course, has been a delayed reflection of the changing use of computer technology. The problem, however, is that now that computers are replacing cameras, telephones, record players, and postal systems it is becoming impossible to attempt to write a comprehensive history of computing that covers all these areas, examines applications and users (which many professionally trained historians of

[28] Of course, in many cases there may not be any archives to draw on, which highlights the importance of historical preservation and explains why the same institutions (e.g., MIT and Metropolitan Life) keep coming up again and again in historical work: historians write about the material found in archives that is well cared for and accessible, and tend to ignore the material that isn't.

technology believe to be vital), and puts the computer into context as an extension of what went before.

Academically trained historians of computing tend to agree that a shift in emphasis from study of the history of hardware to the history of software is intellectually desirable, given the increasing economic and technical importance of software and its vital role in determining the actual social functioning of computer systems. Unfortunately software is much harder to study. Software is less likely to be preserved than hardware; it is produced by many, many times more groups (mostly in end-user organizations); there are fundamental differences among system software, application software, and scientific software; and it is often impossible to separate software from business services or processes. Most attention given so far to the history of software by professionally trained historians has focused on attempts to formalize coding practices through the use of formal methods and the creation of software engineering as a true discipline.[29] These topics loomed large in the minds of credentialed researchers, but do not appear to have been particularly influential on most actual programmers; therefore questions of what ordinary programmers actually did remain almost entirely unaddressed.

Some areas, of course, have been explored much better than others. ACM SIGPLAN organized two major conferences on the history of programming languages, including lengthy articles by the creators of key languages, and discussion of their design and subsequent evolution. These were published in book form.[30] Similar efforts were mounted by those involved by the designers (and in some cases early users) of many important commercial computers such as the IBM 701, IBM 650, and Burroughs 5000, resulting in special issues of *Annals of the History of Computing*. The history of early firms involved in the computer business has been quite well covered, including a remarkably detailed series of technical histories of IBM's work in the 1950s and 1960s. The stories of the first, one-off experimental computers such as the ENIAC, EDSAC, and WHIRLWIND are quite well documented. Much work in these areas consists of the memoirs of participants or of very detailed and narrowly focused accounts of the development of a specific system. While valuable and interesting in their own right, many of these topics are still waiting for professionally trained historians to revisit them and put them into a broader context.

The history of many other important technologies, including operating systems and database management systems, has gone almost undocumented. Indeed, despite the popularity of Microsoft and Bill Gates as journalistic subjects, nobody appears to have tried writing an overview of the history of operating systems, whether scholarly or popular. There are plenty of fun-to-read, journalistic books about the early history of the personal computer industry, including at least one decent-sized shelf devoted entirely to Apple. So far, however, no

[29] The body of work on the intellectual development of software engineering concepts includes Michael S. Mahoney, "Finding a History of Software Engineering," *IEEE Annals of the History of Computing* 25/1 (Jan-Mar 2004), http://www.princeton.edu/~mike/articles/finding/finding.html; Donald MacKenzie, *Mechanizing Proof* (Cambridge, MA, 2001); Stuart Shapiro, "Splitting the Difference: The Historical Necessity of Synthesis in Software Engineering," *IEEE Annals of the History of Computing* 19/1 (Jan-Mar 1997): 20-54; James E. Tomayko, "Software as Engineering," in *Mapping the History of Computing: Software Issues*, ed. Ulf Hashagen, Reinhard Keil-Slawik and Arthur L. Norberg (New York, 2002); Nathan Ensmenger, "From Black Art to Industrial Discipline: The Software Crisis and the Management of Programmers," Ph.D. dissertation (University of Pennsylvania, 2001); and Maria Eloina Pelaez Valdez, "A Gift From Pandora's Box: The Software Crisis," Ph.D. dissertation (University of Edinburgh, 1988).

[30] Richard L Wexelblat (ed.), *History of Programming Languages* (New York, 1981); and Thomas J Bergin and Rick G Gibson (eds.), *History of Programming Languages II* (New York, 1996).

professionally trained historian has ventured more than a chapter on the personal computer era, which has done little to advance our understanding.[31]

Almost nothing has been written about the history of computer science as an academic discipline. The main exceptions are a series of articles on early computer science departments and NSF funding by William Aspray, some short articles on theoretical computer science by Michael Mahoney, and two books on the role of DARPA in funding computer science. There are many different ways in which the topic could be approached, including a conceptual examination of the influx of ideas from different areas of mathematics, institutional histories of key departments, books on the development of ACM and the other relevant academic societies, or biographies of key computer scientists. The ACM has recently made some moves toward support of historical work, which if successful will help in preserving its own history and that of the discipline.

The history of scientific computing is reasonably well served into the mid-1950s, at which point the secondary literature largely gives out. There have been a number of publications covering table-making in the USA and UK and differential analyzers; a flurry of material on Charles Babbage; a good biography of von Neumann; and material on the service organization created by IBM to support its first scientific users. With the exception of one paper on supercomputing and the national labs, the only papers that come to mind for the post-1955 period are retrospectives of major machines and packages written by the creators themselves, often to celebrate the 20th or 25th anniversaries of their introduction.[32]

In short, then, the queue of worthy topics is continually growing longer, as new areas of study develop and pile up more rapidly than the small but doughty band of historians can deal with the old ones. Whatever your personal area of interest, you'll probably find that almost all of it has so far escaped the notice of historians, though you will probably find some interesting and insightful papers on related topics. However, to look on the bright side, an impressive quantity of excellent work has appeared in the last few years, and there are signs that historians are starting to gain an intellectual purchase on some of the foothills of this imposing mountain range.

What to Learn from History

Historian Daniel Boorstein famously wrote that planning for the future without an awareness of the past is like trying to plant cut flowers. This view finds considerable support among historians, who quote it at every opportunity, and very little among technologists. In this they are not so very different from the mainstream of American culture. As a people, white Americans have an unusually abstract relationship to their own history. It's not that Americans downplay history. A pervasive, state-fostered, and stylized mythology of turkey-eating pilgrims, brilliant founding fathers, selfless flag-sewing women, brave guerilla fighters, and ingenious

[31] Historians are just beginning to come to grips with the post-1975 period. The best example so far is probably the discussion of the microcomputer software industry in Martin Campbell-Kelly, *From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry* (Cambridge, MA, 2003).

[32] See the "Scientific Computing" section of my "Key Resources in the History of Computing" for citations and details of the books mentioned here. Not included there are the paper on IBM's first scientific support group by Atsushi Akera, "IBM's Early Adaptation to Cold War Markets: Cuthbert Hurd and His Applied Science Field Men," *Business History Review* 76/4 (Winter 2002): 767-802; and the paper on supercomputing by Donald MacKenzie, "The Influence of Los Alamos and Livermore National Laboratories on the Development of Supercomputing," *IEEE Annals of the History of Computing* 13/2 (April 1991):179-201.

inventors serves to rally patriotic sentiment on national holidays, in preparation for clearance events at the local mall. But while the flag and the constitution retain enormous symbolic power, they resonate very much as symbols of present rather than past glory. In many other places such as Ireland, Greece, or much of the Muslim world, discussion of current grievances and disputes inevitably blends with reference to events taking place hundreds or thousands of years ago. This is not all bad. As an Englishman, I find it reassuring that I can safely attend a 4th of July barbecue or walk across the famous bridge at Concord without anyone feeling any upwelling of "ancient hatreds" toward me. Professionally, however, this pervasive indifference to the substance of the past makes the teaching of any kind of history to non-specialists an uphill struggle.

The Denial of the Past

A few years ago, amid the frenzied hype of the Internet boom, one of the long-time columnists of one of America's most successful personal computing magazines addressed the issue of technological change. He reflected that the wave of technological change brought about by the Internet was entirely unprecedented, and that his own age faced challenges and opportunities unlike any other. His great-grandfather, he suggested, had died in a world very much like the one he was born into. He, on the other hand, had lived through upheavals of ever increasing magnitude. Similar claims are made all the time, but one needs only a very passing acquaintance with history to realize that this is complete rubbish. Imagine an American who was born in 1895 and died in 1970. She would have seen, among many other things, the development of personal transportation from the biological horse to the Ford Mustang, the development of communication from the telegraph to the communications satellite, the development of home entertainment from the player piano to the stereo LP, the development of the film studios, talkies and Technicolor, the creation of television and radio broadcasting and networks, the introduction of electrical refrigerators, the Apollo landings, nylon, antibiotics, the atomic bomb, intercontinental passenger jet service, supermarkets, suburbs and shopping malls, the mainframe and the minicomputer.

Admittedly the half lifetime since 1970 has replaced the LP with the CD; the Boeing 747-100 with, well, the Boeing 747-400; and big malls with enormous malls. It has invented the SUV, and has seen huge drops in the cost of long-distance communication and incremental advances in medical care. But the fact that such a claim could go unchallenged in one of America's best-selling technological publications indicates a depressing triumph of ignorance. If people had been just a little less ignorant of history, they might not have believed that, for example, a grocery home delivery firm like Webvan could be the next Microsoft. For one thing, they would have known that grocery stores used to make deliveries, and that the practice died out with the rise of large supermarkets, increasing population diffusion, and greatly increased ownership of personal automobiles during the first half of the twentieth century. The web browser didn't change those things. History is sometimes defined as the study of change over time, but paradoxically the most valuable thing history can contribute to many areas is an appreciation of continuity.

The computer field is divorced from its own past. Programmers can be viewed as "obsolete" at the age of thirty-five. In the popular view of the past forty years, one technological transition created the computing world we know (always about five years ago) and another one will arrive to fix all the problems with it (sometime next month). It was hard to convince PC programmers that they had much to learn from the software engineering techniques developed for mainframe projects, and it is hard to convince young Java programmers that any of the problems they grapple with might have existed ten years ago. But we know that, on the

technical level at least, every idea, every system (or at least all those in real use), and every architecture is constantly evolving. Without a knowledge of history, we can never understand why things are the way they are.

This lesson is a hard one to convey to students. In one recent class I assigned for my students an excerpt from Ellen Ullman's book, *Close to the Machine*, in which she spoke of the dozens of now-obsolete languages she had used over the course of her career and pondered the mass of soon-to-be-obsolete material arriving with every new shipment from her Microsoft Developer's Library subscription.[33] In a discussion section, the students were asked to discuss their own feelings about entering a field in which they would be required to relearn their entire technical skill-set on a frequent basis. Even after reading the article, few of them believed that this would happen to them. They were confident that the smattering of Java and Windows they had picked up during their education would stay current indefinitely.

Knowledge of the broader history of technology, and indeed of history in general, may be constructive here. History may also help the discipline of computer science ~~a~~nd the related fields of information science, information systems, informatics, and software engineering ~~fi~~gure out what they are. For almost forty years, employers have been complaining that universities produce computer science graduates with bad attitudes and a dearth of useful skills. People in universities have countered with the claim that they give education rather than vocational training, and provide the core skills and understanding that underlie current and future practices. That's how they justify having spent decades teaching Pascal, SML, LISP, and Smalltalk rather than COBOL, Visual Basic, and PERL.

Academics are thus charged to recognize the "timeless principles" in their own fields, knowledge of which will benefit students for many years and through changes in specific technologies. It seems reasonable to suggest that having knowledge of what has remained stable in the past may help us to work out what is liable to remain stable in the future. In my own research, for example, I discovered that many central dynamics of corporate computer use had remained unchanged over its fifty-year history, such as the difficulties of diagramming and describing business processes, the refusal of most developers to use the methodologies favored by academic experts, and the inescapable fact that most computer departments are hated by their users.[34] And, I would suggest, the same is true of the computer science field itself, and of all its various subdisciplines. To know who you are, you must know where we came from. You must know your own story. Why is computer science what it is, and not something else entirely? When the time comes to shift, how will you know? What should you keep, and what will you discard?

Well, enough of the rhetoric and hand-waving. I would suggest, however, that perhaps the biggest contribution that history can make to improving undergraduate education is in improving the understanding that undergraduate educators have of their own field and where it came from. It might never need to show up on the syllabus at all to accomplish that.

Tactical and Strategic Users of History

Let's assume, though, that an instructor or curriculum designer is convinced of the utility of including historical material in a computing course. I'll leave it to the other contributors to talk

[33] Ellen Ullman, *Close to the Machine: Technophilia and its Discontents* (San Francisco, 1997).
[34] Thomas Haigh, "Technology, Information and Power: Managerial Technicians in Corporate America," Ph.D. dissertation, University of Pennsylvania (Philadelphia, 2003).

about exactly how that might be accomplished. I do, however, believe that there is one vital question here, and that each contributor has implicitly come down on one side or another on this question, without necessarily realizing this consciously. That question is whether the contribution of history to computing education should be as a tactical tool designed to help students grasp the same material and build the same skills that they are already receiving, or as a strategic challenge to fundamentally rethink the curriculum design, skills, and culture of computer science education.

One can imagine history playing a useful part on the tactical level either as a pedagogical innovation comparable to the use of PowerPoint, in tutorial sessions, or in on-line discussion groups. It's not hard, for example, to believe that an understanding of the historical development of file management systems and the network data model might help students to appreciate the strengths and weaknesses of the current DBMS systems and of the relational model. Dropping snippets of biography into the class, to let students know who Turing was when they discuss his machine, might help to keep them awake. This might not involve what most academic historians would think of as "good" history. The history would often consist of little, disconnected snippets of factual material, stripped of its richness and complexity, its contingency and context. Instructors might hide details and skip over complexities to create neat parables or inspirational fables, such as the apocryphal but widely repeated story that the former chairman of IBM stated that the world market for computers would be around five, or the story that NASA spent millions to create a space pen while the Russians used pencils. On the other hand, if it helps to teach good computer science, who cares if it is bad history? Purists in any area tend to be shocked at what the masses do with their work.

There are also some areas of computer science where integration of historical material is an obvious tactic, perhaps an inescapable necessity. For example, in a computer architecture or operating course it is hard to begin with discussion of current systems because they are enormously complex. Many instructors would start with simple, manageable systems (such as Andrew Tanenbaum's MINIX, an instructional recreation of early UNIX), perhaps recapitulating during the progression of their courses some of the same evolution toward increasingly complex systems that has taken place over history. Perhaps better history might help to produce better texts, stressing the specific factors influencing different designs, just as business schools use detailed case studies to teach hard-to-formalize kinds of skill.[35] Likewise, theoretical computer science might follow mathematics in presenting a carefully ordered sequence of developments in which each new topic rests on the proofs presented earlier in the course, and so resembles a rather stylized and formal kind of history.

How about the idea that incorporating history in the teaching of computing would fundamentally shake up our assumptions about what it is that students should be learning? This position appeared to be held, consciously or subconsciously, by almost all the professionally trained historians at the workshop and by none of the practicing computer scientists. As one historian put it, "We should be training them to think like historians." What might this mean in practice?

It is important to realize that undergraduates are not usually taught history because anybody believes that the factual content of a course on, for example, nineteenth century

[35] MINIX is described in Andrew S. Tanenbaum and Albert S. Woodhull, *Operating Systems: Design and Implementation, 2nd Edition* (Englewood Cliffs, 1997). An important operating systems text with a developmental perspective is Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, *Operating System Concepts,* 5th Edition (New York, 1998).

diplomatic history will be directly useful to them in their work lives. There are certainly specific lessons there for policymakers dealing with current challenges, but history is not marketed primarily as vocational training for future State Department officials. The most crucial skill for a history major to develop is not necessarily the craft of tracking down obscure historical sources, but rather that of interpreting the biases and assumptions of their authors and fitting them together into a coherent whole.

For example, in 1998, and again in 2001, I taught a course on the idea of computer technology and social revolution, designed to help students get a handle on the overblown claims being made for the social impacts of the Internet by looking at previous technologies, particularly the mainframe in the 1950s and the PC in the 1970s and 80s, but also apparently unrelated technologies such as nuclear power, ham radio, and the telegraph. Students examined these earlier technologies, and saw many kinds of similarity with current ones (from the utopian claims made for them, to the hobbyist cultures that developed around them, to the alarmist warnings on their threats to traditional values). Another course mixed historical and recent examinations of the role of information in structuring companies, markets, and customer relationships in business. These courses gave the students a perspective to deal more critically with technological hype, not just for dot-com stocks (which vanished without any real help from me), but with future waves of technological mania as yet undreamt of.[36]

In a seminar course of this kind, intensive group discussion of source material is vital. The source material is not in itself the most important thing. In my teaching I've assigned the film *Wargames*, overheated extracts from journalistic articles on hackers, science fiction stories from the 1950s, contemporary reports on the failed videotext networks of the 1980s, and utterly dated predictions from the 1970s about the future of personal computing. None of these pieces of writing is assigned purely for its factual content (which is dubious), literary merit (often limited), or intellectual rigor (sadly lacking). Rather the point is to get students to examine them critically, and uncover the kinds of social influence and unexamined assumptions that we often cannot see in contemporary materials because we are too close to them. With liberal arts students at an elite private college, and a class size of around a dozen, the course was a reasonable success (at least the students gave good evaluations and enjoyed it, even if they didn't always do all the reading).

As other contributors to this volume demonstrate, one faces an enormous hurdle in trying to use techniques of this kind with computing students. Computing students tend to assume that courses will provide a large volume of factual technical information, which the majority would like to be of direct and obvious relevance to potential employment. They expect to cram this material into their head shortly before an examination, working from some kind of bullet-point list. They are not necessarily predisposed to respect the kind of "fuzzy" or "touchy-feely" activities valued by historians, such as group discussion, evaluation of arguments made by others, or the analysis of technical systems within a broader social context. If they had wanted to do those things, they probably wouldn't have picked the major they did. The particular problem is that computing students are disinclined to accept that study of particular examples now remote in time and space will teach them much of value.

The other obvious problem with computer science instructors trying to use history to help students think more like historians is that computer science instructors themselves do not know how historians think (any more than most historians understand how computer scientists think).

[36] Syllabi for my old courses are available from http://www.tomandmaria.com/Tom/teaching.

Overcoming this problem would require a profound and long-term realignment of the two disciplines.

History and the Core Curriculum

Yet the strategic potential is real, if hard to tap. Consider the current (2001) version of the joint ACM/IEEE-CS model curriculum for computer science education. History is conspicuous by its absence. Admittedly, the core curriculum does include history as a topic, but it allocates exactly one hour to it (of the sixteen devoted to "social and professional issues"), during which time students are expected to cover "prehistory—the world before 1946," "history of computer hardware, software, networking" and "pioneers of computing." That must make for a rather packed hour and, I fear, a confused and confusing one.[37]

On the other hand, if we look at the five "characteristics a successful graduate should possess" at the end of a computer science program, provided elsewhere in the same report, we see enormous potential for history. Of these five key areas, at least three can be addressed with history.[38]

Adaptability: If we can give students a sense of the unfolding of history and the many profound shifts in technology over the past few decades, they will be much less likely to assume that the current technology is the be-all and end-all. Less experienced practitioners often have a sense of only one generation of technology before the current one, and a foolish idea that if everything can just be migrated to the latest paradigm and really "done right," then it will last forever. A better historical understanding means that students are: 1) less likely to be trapped as specialists in single technologies; 2) more likely to understand the need for flexibility and modularity in design; and 3) less likely to throw away old but serviceable systems and embark on pointless migrations.

System-level perspective: The same factors are at work here. An understanding of the evolution of any real system would expose students to the historical layering of technologies and interfaces. They would be better equipped to maintain code (something CS education does little to equip students for) and might be better able to develop a sense of WHY we must change some parts of a system and leave others alone, or why re-implementation is not always the answer.

Appreciation of the interplay between theory and practice: History teaches us that there are always tradeoffs between resource usage—including of course the resources of time, risk, maintainability, cost, and skill, as well as processor memory and disk usage. Many of these tradeoffs have shifted profoundly with time as hardware costs have dropped spectacularly. Yet, as we all know, students may place an emphasis on optimizing code that makes no sense whatsoever, given the profusion of other resources and the need to lower cost and increase maintainability. How many database students learn when *not* to normalize? Some of what we teach students may reflect historical conditions that no longer hold. To generalize this a little, many techniques have niches within which they work well. The hardest thing is to teach students when each is appropriate, and when it should not be used. History can help us here.

[37] http://www.computer.org/education/cc2001/final/sp.htm#SP-ProfessionalResponsibility.
[38] http://www.computer.org/education/cc2001/final/chapter11.htm.

Conclusion

Let me finish, then, on this note of hope. Professionally trained historians are taking an increasing interest in the history of computing, and are working cooperatively with interested computer scientists and business and scientific computing pioneers. A diverse and growing literature covers the history of computing from many different perspectives, albeit far from comprehensively. There are real opportunities for the incorporation of history into the core computer science curriculum, in both tactical and strategic ways. More immediately, having knowledge of the history of your own field will make you a better teacher and a better researcher.

Acknowledgments: While this paper has the same basic structure as that presented at the 2001 Amherst workshop, "Using History to Improve Undergraduate Education in Computer Science," I have been quite shameless in plundering good ideas and acute observations presented elsewhere during that workshop, and in the two smaller events held during 2004 to support the creation of this volume. The contributions of William Aspray, Martin Campbell-Kelly, Atsushi Akera, Michael Mahoney, and Thomas Drucker were particularly important in shaping my thinking.

Thomas Haigh is an assistant professor in the School of Information Studies at the University of Wisconsin, Milwaukee. Haigh holds a Ph.D. from the University of Pennsylvania, and has published articles on the history of the software industry, the origins of the database management system, the historical relationship of computing to corporate management, and the creation of the data processing department. His current research projects include the social history of the personal computer and the origins of open source software in the scientific computing field. He serves as Biographies Editor of *IEEE Annals of the History of Computing*, and is a member of the Committee on ACM History. *Author's address*: SOIS, Bolton Hall, Room 568, P.O. Box 413, Milwaukee, WI 53201 USA. Website: www.tomandmaria.com/tom. Email: thaigh@acm.org.