Communities of Computing

Computer Science and Society in the ACM

Thomas J. Misa

Communities of Computing

ACM Books

Editor in Chief

M. Tamer Özsu, *University of Waterloo*

ACM Books is a new series of high-quality books for the computer science community, published by ACM in collaboration with Morgan & Claypool Publishers. ACM Books publications are widely distributed in both print and digital formats through booksellers and to libraries (and library consortia) and individual ACM members via the ACM Digital Library platform.

Communities of Computing: Computer Science and Society in the ACM
Thomas J. Misa, Editor, *University of Minnesota*
2017

Text Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining
ChengXiang Zhai, *University of Illinois at Urbana–Champaign*
Sean Massung, *University of Illinois at Urbana–Champaign*
2016

An Architecture for Fast and General Data Processing on Large Clusters
Matei Zaharia, *Massachusetts Institute of Technology*
2016

Reactive Internet Programming: State Chart XML in Action
Franck Barbier, *University of Pau, France*
2016

Verified Functional Programming in Agda
Aaron Stump, *The University of Iowa*
2016

The VR Book: Human-Centered Design for Virtual Reality
Jason Jerald, *NextGen Interactions*
2016

Ada's Legacy: Cultures of Computing from the Victorian to the Digital Age
Robin Hammerman, *Stevens Institute of Technology*
Andrew L. Russell, *Stevens Institute of Technology*
2016

Edmund Berkeley and the Social Responsibility of Computer Professionals
Bernadette Longo, *New Jersey Institute of Technology*
2015

Candidate Multilinear Maps
Sanjam Garg, *University of California, Berkeley*
2015

Smarter than Their Machines: Oral Histories of Pioneers in Interactive Computing
John Cullinane, *Northeastern University; Mossavar-Rahmani Center for Business and Government, John F. Kennedy School of Government, Harvard University*
2015

A Framework for Scientific Discovery through Video Games
Seth Cooper, *University of Washington*
2014

Trust Extension as a Mechanism for Secure Code Execution on Commodity Computers
Bryan Jeffrey Parno, *Microsoft Research*
2014

Embracing Interference in Wireless Systems
Shyamnath Gollakota, *University of Washington*
2014

Communities of Computing

Computer Science and Society in the ACM

Thomas J. Misa, Editor

University of Minnesota

ACM Books #13

Communities of Computing

Thomas J. Misa, Editor

books.acm.org
www.morganclaypoolpublishers.com

ISBN: 978-1-97000-187-7 hardcover
ISBN: 978-1-97000-184-6 paperback
ISBN: 978-1-97000-185-3 ebook
ISBN: 978-1-97000-186-0 ePub

Series ISSN: 2374-6769 print 2374-6777 electronic

DOIs:

10.1145/2973856 Book
10.1145/2973856.2973857 Preface
10.1145/2973856.2973858 Chapter 1
10.1145/2973856.2973859 Chapter 2
10.1145/2973856.2973860 Chapter 3
10.1145/2973856.2973861 Chapter 4
10.1145/2973856.2973862 Chapter 5
10.1145/2973856.2973863 Chapter 6
10.1145/2973856.2973864 Chapter 7
10.1145/2973856.2973865 Chapter 8
10.1145/2973856.2973866 Chapter 9
10.1145/2973856.2973867 Chapter 10
10.1145/2973856.2973868 Chapter 11
10.1145/2973856.2973869 Chapter 12
10.1145/2973856.2973870 Chapter 13
10.1145/2973856.2973871 References

A publication in the ACM Books series, #13
Editor in Chief: M. Tamer Özsu, *University of Waterloo*
Area Editor: Thomas J. Misa, *University of Minnesota*

First Edition

10 9 8 7 6 5 4 3 2 1

Contents

Preface xi

Chapter 1 **ACM and the Computing Revolution** 1

Thomas J. Misa

1.1 History in Computing 6
1.2 History of Computing 10
1.3 Chapters of ACM History 13

DEFINING THE DISCIPLINE 25

Chapter 2 **From Handmaiden to "Proper Intellectual Discipline": Creating a Scientific Identity for Computer Science in 1960s America** 25

Janet Abbate

2.1 Introduction 25
2.2 The Status of Computing at NSF in the Early 1960s 31
2.3 Organizational Boundary-work: Getting a Seat at the Table 32
2.4 Discursive Boundary-Work: Establishing a Scientific Identity 36
2.5 Success: The Creation of NSF's Office of Computing Activities 42
2.6 Conclusion 45
2.7 Acknowledgments 46

Chapter 3 **George Forsythe and the Creation of Computer Science As We Know It** 47

Joseph November

3.1 The Man Who Would Remove the "M" from the ACM 47
3.2 Forsythe Before CS: From Mathematics to Meteorology to Computing 50

3.3 Hard Lessons on the Road to Computer Science 51
3.4 Building a Home for Computer Science at Stanford 56
3.5 Forsythe and the Challenge of Defining “Computer Science” 58
3.6 Conclusion 70

Chapter 4 **Solving a Career Equation: The First Doctoral Women in Computer Science** 71

Irina Nikivincze

4.1 Introduction 71
4.2 Historical Context 73
4.3 Gender and Science 74
4.4 Gender and Computing: Identifying the Problems 76
4.5 Data and Method 77
4.6 Findings 79
4.7 Conclusion 89

Chapter 5 **The History and Purpose of Computing Curricula (1960s–2000s)** 91

Sebastian Dziallas and Sally Fincher

5.1 Always Volunteers: Coordinating Volunteer Efforts 92
5.2 Accreditation: Addressing a Threat to the Reputation of the Discipline 96
5.3 Definition of a Discipline 101
5.4 Conclusion 108
5.5 Acknowledgments 109

BROADENING THE PROFESSION 111

Chapter 6 **“Deeply Political and Social Issues”: Debates within ACM 1965–1985** 111

Janet Toland

6.1 Introduction 111
6.2 1969: A Question of Importance 114
6.3 1972: The Equal Rights Amendment 116
6.4 1975: The Turchin Issue 119
6.5 The Committee on Computers and Public Policy 122
6.6 History of SIGCAS 125
6.7 Analysis of Changing Interests within SIGCAS 133
6.8 Conclusion 135
Appendix A 138

Chapter 7 **Organized Advocacy for Professional Women in Computing: Comparing Histories of the AWC and ACM–W** 143

Amy Sue Bix

7.1 Introduction 143
7.2 Gender, Computing, and Organized Advocacy for Women 150
7.3 Advocacy for Women in Computing and K-12 Outreach 157
7.4 The Shifting Nature of Advocacy for Women in Computers 162
7.5 Conclusion 171

Chapter 8 **The Development of Computer Professionalization in Canada** 173

Scott Campbell

8.1 Rise of Canadian Computing 174
8.2 ACM's Early Start in Canada 176
8.3 Canadian Computing and Data Processing Societies 179
8.4 The DPMA in Canada 184
8.5 CIPS, DPMA, and the Canadian Accreditation Battle 189
8.6 Toward Canadian Identity? 195

Chapter 9 **The Anatomy of an Encounter: Transnational Mediation and Discipline Building in Cold War Computer Science** 199

Ksenia Tatarchenko

9.1 Novosibirsk—Moscow—New York—San Francisco— Los Angeles 203
9.2 Person-to-Person, Institution-to-Institution, Discipline-to-Discipline 210
9.3 Twisted Truths: Dealing with Hazards of Boundary-Crossing 218
9.4 Conclusions: Divided Worlds, a Shared Community 226

Chapter 10 **Concern for the 'Disadvantaged': ACM's Role in Training and Education for Communities of Color (1958–1975)** 229

R. Arvid Nelsen

10.1 Introduction 229
10.2 The Discussion of Social Implications and Issues within the ACM 232
10.3 Programs for the "Disadvantaged"—1968–1972 240
10.4 ACM Involvement 248
10.5 ACM Establishes a National Committee on Computing and the Disadvantaged 252
10.6 Conclusion 257

EXPANDING RESEARCH FRONTIERS 259

Chapter 11 **Other Places of Invention: Computer Graphics at the University of Utah** 259

Jacob Gaboury

11.1 Introduction 259
11.2 Salt Lake City, 1966 263
11.3 Practical Applications 267
11.4 Problem Solving 272
11.5 Community 277
11.6 Other Places 284

Chapter 12 **Framing Computer Security and Privacy, 1967–1992** 287

Rebecca Slayton

12.1 Framing Record-Keeping Security 292
12.2 Transitions in the 1980s: Specialization and the Growth of Computer Networking 311
12.3 Reframing Security Amid Growing Computer Networking 313
12.4 Conclusion 328

Chapter 13 **Hypertext, Digital Libraries, and Beyond: A History of ACM SIGWEB** 331

Inna Kouper

13.1 Introduction 331
13.2 The Association for Computing Machinery (ACM) 333
13.3 ACM SIGWEB 335
13.4 Conclusion: Toward a Model of Epistemic Work in Professional Organizations 349
Acknowledgments 353

References 355

Index 381

Contributor Biographies 405

Preface

I like to think the origins of this volume stretch back to a visit that archivist Bruce Bruemmer made to Edmund Berkeley at his home in Newton, Massachusetts. Berkeley was well along in a busy and tumultuous life, as an author, activist, publisher, computer visionary, and co-founder of the Association for Computing Machinery. By my reckoning, he had then published three dozen books. He is of course most famous for the popular and readable *Giant Brains, or Machines That Think* (1949), but his first major work was published more than a decade earlier in 1937 by the American Institute of Actuaries. It was a 40-page treatise bearing the unwieldy title of "Boolean algebra (the technique for manipulating 'and', 'or', 'not', and conditions) and applications to insurance." Remarkably, it was promptly reviewed in the *Journal of Symbolic Logic* by none other than Alonzo Church. Church was the eminent Princeton mathematician who, just a few months earlier, had recognized the unusual insight contained in a paper on computable numbers by a young Alan Turing (three years Berkeley's junior). In his review, Church praised Berkeley's exposition of Boolean algebra "with illustrative examples worked out in detail" and connected to insurance applications. He favorably compared Berkeley's examples to those given by logician John Venn. Church closed with a warm observation, noting the work's "novelty lies in the practical application of this technique in a hitherto unsuspected direction."[1]

I imagine that archivist Bruemmer might have entertained dreams of finding faded correspondence with Church or Venn lurking in Berkeley's basement, or perhaps page proofs from *Giant Brains,* or possibly his proposal to computerize Prudential Insurance, likely a note from Grace Murray Hopper from their Harvard Mark I days, or even a snippet from Berkeley's indefatigable anti-war activism. Surely he

1. E. C. Berkeley. 1937. Boolean algebra (the technique for manipulating 'and', 'or', 'not', and conditions) and applications to insurance. *The Record of the American Institute of Actuaries*, 26, part 2, (54): 373–414; reviewed by A. Church. 1938. *J. Symb. Logic*, 3(2): 90.

hoped for insight on the early days of ACM. But that day, as Bruemmer later related the story, "he assured me that he had no records." In the archival business, there's no room for impatience. Bruemmer continued his unwearied contacts and was rewarded when, after Berkeley died in 1988, "I personally carried over 100 boxes out of his basement."[2] This was the first significant cache of ACM-related archival records that made their way into a publicly accessible archive; in subsequent years, the Charles Babbage Institute assembled records from ACM officers and awardees culminating with the transfer in 2008 of the ACM headquarters records.

Close study of historical records demands that there be records that are accessible and worthwhile for the historian's attention. In this respect, the first set of acknowledgments for this volume on ACM history is to the donors of valuable historical records as well as the archivists who sought them out. History draws eclectically on many different kinds sources, including photographs, artifacts, oral histories, and published accounts. But the bedrock for historical analysis is the rich detail provided by archival records: the day-by-day, or month-by-month set of correspondence, memos, committee reports, and the like that give deep and particular insight into the past.[3]

A second category of acknowledgments goes to the far-seeing ACM members who created a designated ACM History Committee in 2004. David Wise, Richard Snodgrass, and others used their understanding of ACM to lodge history in the organization's ongoing operation. Since then, the History Committee has commissioned oral history interviews, expanded the A. M. Turing Award website, and sponsored research fellowships that are the immediate cause of the scholarship assembled in this volume. It's been a personal pleasure to serve as chair of this committee and to work with Mary Hall, in one transition of the committee chair, as well as with in-coming History Committee chair Chuck House. The History Committee supported the research appearing in this volume as well as a special workshop at the 2015 annual meeting of the Society for the History of Technology and its history of computing interest group SIGCIS. Many of the contributors benefitted from presenting their research and getting valuable audience feedback in Albuquerque, New Mexico.

2. B. H. Bruemmer and E. Kaplan. 2001. Realizing the concept: A history of the CBI Archives. *IEEE Ann. Hist. Comput.*, 23(4): 29–38, quote on p. 31.

3. For additional ACM archival records, see ACM Research Materials posted at http://history.acm.org/content.php?do=links; also see T. Haigh, E. Kaplan, and C. Seib. 2007. Sources for ACM history. *Commun. ACM*, 50(5): 36–41.

Finally, at the Charles Babbage Institute, I have a number of personal acknowledgments. To begin, I am periodically amazed at the archival treasures on ACM and other computing topics collected over the years by the succession of professional archivists. Bruce Bruemmer collected the Berkeley papers; his successors including Beth Kaplan, Karen Spilman, Carrie Seib, Stephanie Crowe, and Arvid Nelsen have significantly augmented CBI's holdings of ACM related materials. Arvid also followed up his curiosity about ACM's role in educational programs and contributed one of the chapters in this book. Associate director Jeffrey Yost took time from his work to give sage counsel and valuable suggestions. And once again I owe a special thanks to Katie Charlet who assisted capably with the production of this book manuscript (and kept other parts of CBI so well organized that I imagine sometimes that it's a machine that would run itself—except that I know better!).

With the ACM Books venture, I have enjoyed shaping this volume through conversations with editor-in-chief Tamer Özsu, who enthusiastically supported a book on ACM history (once he was assured that this was no unedited conference volume), and with Morgan & Claypool's Diane Cerra and her capable production staff. We have the hope, unusual for a historical volume, of having actual publication within 12 months from the SHOT–SIGCIS conference papers that were the origin of several of these chapters. With good patience and close attention to the calendar, authors have now endured three rounds of revisions, and at last they have my word that this will be the "last" "final" version.

Thomas J. Misa
ACM History Committee (chair 2014–2016)

1 ACM and the Computing Revolution

Thomas J. Misa

The Association for Computing Machinery and the computing revolution grew up together, and this book is the first to bring together essential chapters in this remarkable story. It begins in the year 1946, a seminal moment in computing, when the legion of secret wartime efforts in radar, cryptography, advanced electronics, and high-speed communication—on both sides of the Atlantic—began to gel as a public movement to foster the emerging field of digital electronic computing. Britain's wartime proto-computers at Bletchley Park were still under the tight wrap of state security, and Germany's leading wartime effort was buried by a bomb in Berlin. The United States, with its cities undamaged and its economy intact, accordingly took an early lead in computing. In February 1946 the U.S. Army staged a high-profile public display of the ENIAC, a vacuum-tube wonder built at the University of Pennsylvania's Moore School of Engineering that is frequently hailed at the world's first electronic computer. That spring the two men most responsible for the technical design of ENIAC left the University of Pennsylvania to found what became the much-storied Univac company. A few months later in the summer, 50 or so of the world's computing experts spent eight weeks in Philadelphia at the Moore School learning about ENIAC and the exciting plans for a successor machine called EDVAC that promised true programmability.[1]

The computing community soon expanded to several hundred engineers, mathematicians, military officers, scientists, and a few notable pioneer programmers, who assembled in the spring of 1947 at notable computing conferences at Harvard

1. F. M. Verzuh. Moore School of Electrical Engineering lecture notes (July 8–August 31, 1946), Charles Babbage Institute (CBI 51), http://purl.umn.edu/41379; T. Haigh, M. Priestley and C. Rope. 2016. *ENIAC in Action: Making and Remaking the Modern Computer.* MIT Press, Cambridge, MA.

and at MIT, as well as a meeting convened by the New York section of the American Institute of Electrical Engineers (AIEE). That fall in mid-September, Columbia University hosted the inaugural meeting of the "Eastern Association for Computing Machinery" where it proclaimed a lofty aim to "advance the science, development, construction, and application of the new machinery for computing, reasoning, and other handling of information."[2]

The "principal person in the founding of ACM" was Edmund Berkeley, an inveterate organizer, gadfly, and computer enthusiast, soon to publish a popular book with the provocative and anthropomorphic title *Giant Brains, or Machines That Think* (1949), and he brought together 57 people for that inaugural meeting at Columbia.[3] Within a few months, as soon as enough members from across the country signed up, the qualification "eastern" was dropped and the simply named Association for Computing Machinery expanded to 400 members. Frances Holberton, one of the original six ENIAC programmers, recalled that ACM's founding was "one of the things that started the ball rolling . . . the beginning of everybody talking to each other."[4]

With computers the size of offices, and the entire world's computing community then able to fit inside a modest high-school gymnasium, it was at the time impossible to see the future whose foundations were being formed. Today, with computers small enough to put into your pocket—even if the largest floors of web servers or supercomputing complexes are still of immense physical size—the ACM membership numbers 100,000 members around the world. Its largest special interest groups, or SIGs, are each the size of a respectable professional society. Its flagship journal, *Communications of the ACM*, is the go-to publication for monthly news about the computing profession and far-reaching technical advances. And the ACM's Turing Award, named for Alan Turing and first bestowed in 1966 to Alan Perlis, is widely acknowledged to be computing's "Nobel Prize."

In the seven decades between those founding events and today, it goes without saying that the landscape of computing has been transformed. Desktop and laptop computers alone are estimated to consume a 3% share of North America's entire

2. A. Akera. 2007. Edmund Berkeley and the origins of ACM. *Commun. ACM*, 50(5): 30–35. DOI: 10.1145/1230819.1230835; B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals.* Association for Computing Machinery and Morgan & Claypool, New York.

3. P. J. Denning. 1988. Edmund C. Berkeley—ACM founder. *Commun. ACM*, 31(6): 781–782.

4. UNIVAC Conference, Charles Babbage Institute, May 17–18, 1990, Washington, DC, Charles Babbage Institute, OH 200, quotes pp. 100–101. Available at http://purl.umn.edu/104288.

Figure 1.1 **Computer visionary and ACM founder Edmund Berkeley.** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

electricity supply, with giant data centers and server farms taking at least as much or more. The computer-based world of video gaming has spawned an industry that is larger than the movie and music industries combined. Virtual and augmented realities along with the promises of artificial intelligence and the internet of things are bringing about deep social and cultural changes. The recurrent concerns about the automation of muscle and brain work speak also to the economic and cultural

Figure 1.2 **Harvard Mark 3 computer as anthropomorphic "giant brain," following Berkeley.** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

impact of computing. We are still responding to a brave new world where, by some estimates, an iPhone 4 in your pocket has roughly the processing muscle of a 1980s Cray 2 supercomputer (respectively, 1.6 and 1.9 giga-flops).[5]

Besides the technical landscape, the institutional landscape of computing has been transformed; computing directly engages a large share of the world's professional and technical brainpower. For organizing this brainpower ACM's chief competitor today is the IEEE Computer Society, a successor to that early New York meeting of the AIEE. These two societies—ACM and IEEE Computer Society—tower over the more specialized or regional-based computing societies. They are the long-term winners in defining computing as a professional enterprise and scientific field. Several other organizations were for a time undeniably successful. These include the Data Processing Management Association (DPMA), with a huge membership from 1962–1996 when it was reorganized as the Association of Information Technology Professionals (AITP), still active today. The American Federation of Information Processing Societies (AFIPS), in which ACM cooperated with the American Institute of Electrical Engineers (AIEE) and Institute of Radio Engineers (IRE), ran the most important national computer conferences for 25 years (1962–1987) and its international parent, the International Federation for Information Processing (IFIP), is active today and, indeed, figures in several chapters in this book. AFIPS also took a prominent role in shaping the early history of computing by sponsoring *Annals of the History of Computing*, the field's preeminent scholarly journal, and for a time supporting the Charles Babbage Institute, a leading computer history and archiving center. (AIEE merged with IRE to form the IEEE in 1963 and its computing activities evolved to form the IEEE Computer Society in 1971.)

Several other professional and scientific organizations remain prominent in computing today. These include the Society for Industrial and Applied Mathematics (SIAM), founded in 1952, the same year as the predecessor to the DPMA held its first annual conference; the Canadian Information Processing Society, tracing its origins to 1958 and examined in Chapter 8 of this book; and a bevy of organizations founded in the 1970s: the Computing Research Association, a network of the leading computer science departments in North America; USENIX, originally a UNIX users group and today a sponsor of lively conferences; and the Association for Women in Computing which, along with the Anita Borg Institute for Women in

5. L.-B. Desroches, et al. 2014. Computer usage and national energy consumption: Results from a field-metering study, Lawrence Berkeley National Laboratory, December 1. Available at http://eetd.lbl.gov/sites/all/files/computers_lbnl_report_v4.pdf; Processing Power Compared at http://pages.experts-exchange.com/processing-power-compared (accessed June 2016).

Figure 1.3 **ACM membership booth at Spring Joint Computer Conference (c. 1978).** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

Technology and the ACM's Committee on Women in Computing, among others, are key advocates for women in computing. The Anita Borg Institute and ACM are partners in producing the annual Grace Hopper Celebration of Women in Computing (see Chapter 7).

1.1 History in Computing

Despite a common perception that computer scientists lack historical awareness, history featured surprisingly large within ACM long before it founded a designated History Committee in 2004. The periodic History of Programming Languages conferences (1978, 1993, 2007) were high-profile efforts organized by ACM's special-interest group in programming languages, or SIGPLAN. The "prime mover" behind the HOPL effort according to Tim Bergin was Jean Sammet (see Figure 6.1), who as early as 1966 had published "The Use of English as a Programming Language" in

Communications of the ACM. Sammet, already known as the developer of FORMAC and a leading figure in SIGPLAN, went on to write a pioneering book, *Programming Languages: History and Fundamentals* (1969), as well as serve as ACM president (1974–1976). Intriguingly, Sammet's early computing career can be traced to her work for Metropolitan Life Insurance Company in the 1950s, a neat parallel to Edmund Berkeley's work for Prudential Insurance. It is worth pointing out that Bergin was not only a participant in (and historian of) the HOPL conferences; later he was also editor-in-chief of *Annals of the History of Computing*.[6]

The recipe for HOPL's success with these conferences was simple: assemble a notable gallery of well-known programming-language developers, ask them to carefully structure their presentations, let them relate their captivating histories, and leaven the event with cameo appearances by such figures as Grace Hopper and Fred Brooks. The inaugural HOPL featured 13 such languages with talks by John Backus on FORTRAN, Alan Perlis and Peter Naur on ALGOL, John McCarthy on LISP, and Jean Sammet on COBOL, among other luminaries; the conference closed by announcing the launch of *Annals of the History of Computing*. HOPL 2 featured Niklaus Wirth on PASCAL, Barbara Liskov on CLU, Alan Kay on Smalltalk, Dennis Ritchie on C, and Bjarne Stroustrup on C++. HOPL 2 also featured input and participation by Michael Mahoney, the Princeton historian of mathematics and computing who served as conference historian. Mahoney presided over a closing panel—"The History of Programming: Does Our Present Past Have a Future?"—designed to elicit greater awareness of the different types of and diverse viewpoints on history.[7] Some of HOPL's featured languages, even though carefully chosen and deemed central at the time, are not so well known today; for HOPL 1 the language

6. J. E. Sammet. 1966. The use of English as a programming language. *Commun. ACM*, 9(3): 228–230. DOI: 10.1145/365230.365274; T. J. Bergin. 2007. A history of the history of programming languages. *Commun. ACM*, 50(5): 69–71. DOI: 10.1145/1230819.1230841; History of Programming Languages Conference Records, Charles Babbage Institute (CBI 19), http://purl.umn.edu/40668. For a deeper history, see D. Nofre, M. Priestley, and G. Alberts. 2014. When technology became language: the origins of the linguistic conception of computer programming, 1950–1960. *Technol. Cult.*, 55(1): 40–75.

7. Mahoney played a key role in developing *Annals of the History of Computing* as a scholarly publication. His key publications are collected in T. Haigh, editor. 2011. *Histories of Computing.* Harvard University Press, Cambridge, MA. Mahoney's HOPL panel echoes a programmatic essay by historian of science Arnold Thackray, "Science: Has Its Present Past a Future?" in R. H. Stuewer, editor. 1970. *Minnesota Studies in the Philosophy of Science*, vol. 5, pp. 112–127. University of Minnesota Press, Minneapolis, MN. See the HOPL materials in Michael S. Mahoney Papers, Charles Babbage Institute (CBI 213), http://purl.umn.edu/92154 and W. Aspray. 2014. Michael Sean Mahoney (1939–2008), *IEEE Ann. Hist. Comput.*, 36(3): 70–79.

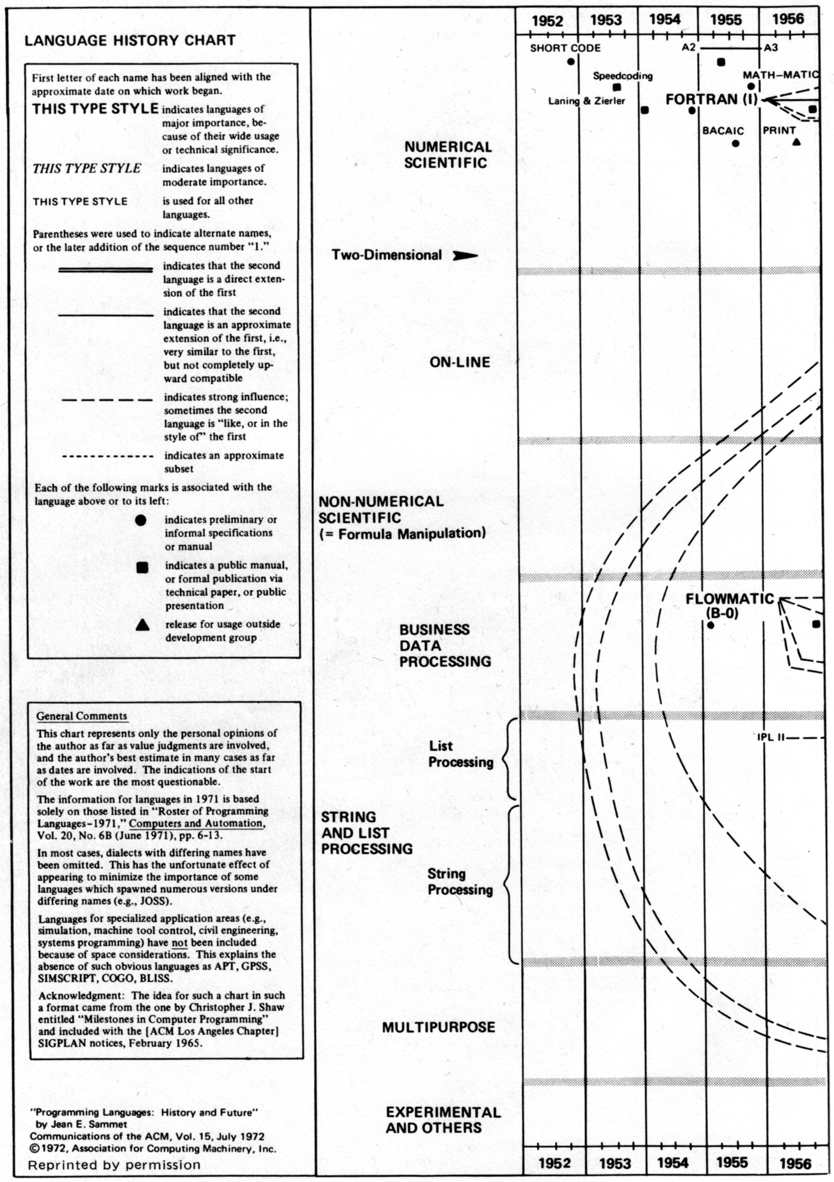

Figure 1.4 **Timelines of Programming Languages from HOPL I.** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

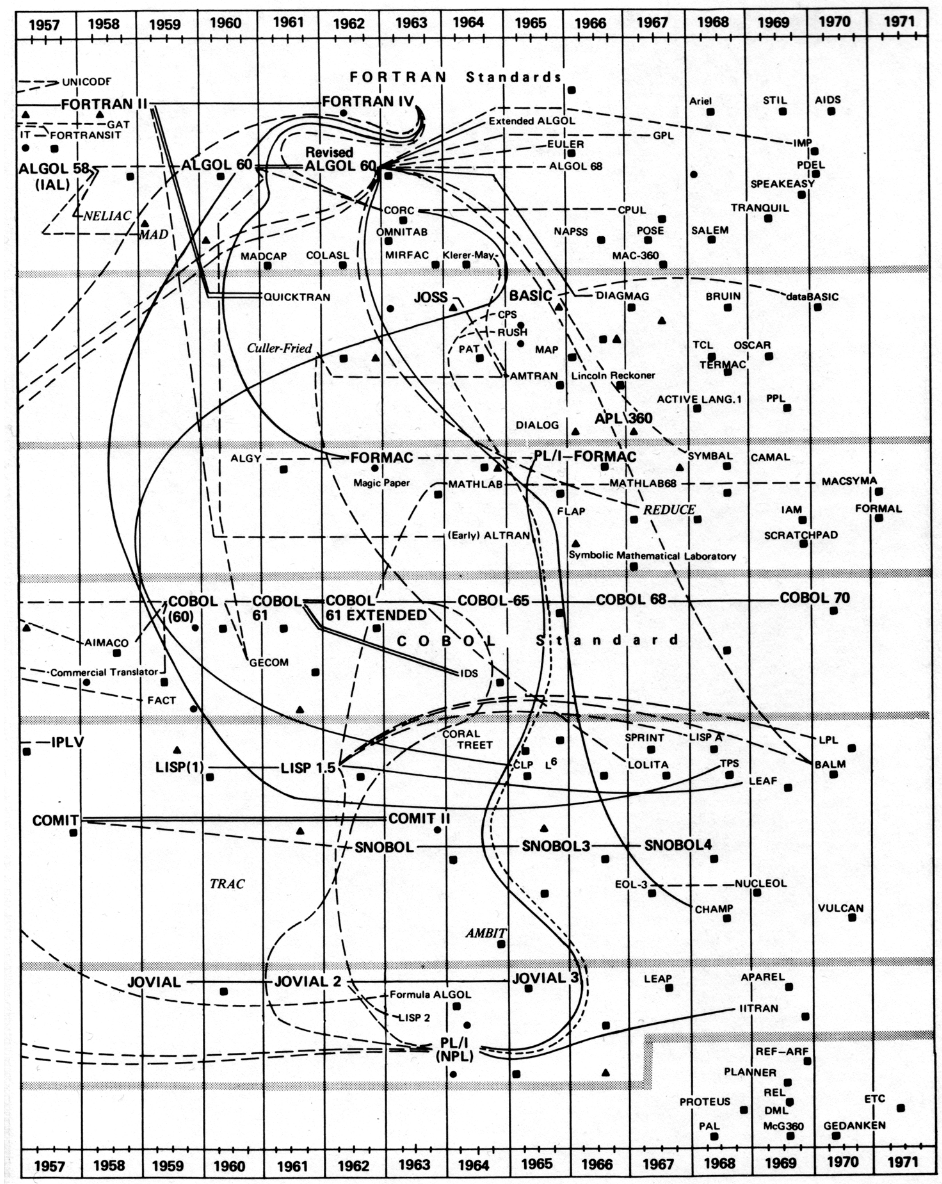

Figure 1.4 *(continued)*

JOVIAL was chosen as an outgrowth of ALGOL 58 and as the "first language to include significant numerical, logical, and data handling capabilities." This is a gentle reminder that history is always written at the present moment, whenever that might be, and reflects the concerns and perspectives of that moment.

Another of ACM's larger special interest groups that actively cultivated its history was SIGGRAPH, which was organized in 1969 by Andy van Dam and held periodic conferences beginning in the 1970s with a notably successful conference in San Jose in 1977 that brought together academic researchers with the emerging graphics industry (see Chapter 11). Doing the math on SIGGRAPH's 25th anniversary in *1998*, you get a ready illustration of the varied perspectives on history: while SIGGRAPH was organized as a full-blown special-interest group in 1969, and its 1977 conference was a breakthrough moment, Carl Machover and the other SIGGRAPH organizers of the 25th anniversary chose instead the origin to be the 1973 conference, a signal that the first in a series of annual conferences is often understood as an origin moment in computing.[8]

1.2 History of Computing

The history of computing as a distinct field with scholarly activity and practitioner interest also took form during the HOPL years. As noted above, historian of science Michael Mahoney played a key role in connecting the HOPL conferences with the community of professional historians. Mahoney, originally a historian of science at Princeton University and author of a biographical study of mathematician Pierre Fermat, was in his career increasingly drawn toward the history of computing and the history of technology. Another practitioner–historian who shaped the field was the University of Michigan's Bernie Galler, the first editor of *Annals of the History of Computing*. (Sponsorship of the journal passed to the IEEE Computer Society in 1992.[9]) Moreover, Galler authored one of two finalist proposals for hosting of the Charles Babbage Institute, which had been organized by Erwin Tomash and associates in California in the late 1970s as a research and archiving center. The winning proposal to host Charles Babbage Institute (CBI), as it turned out, was

8. See files on SIGGRAPH '98 in Carl Machover Papers, Charles Babbage Institute (CBI 206), http://purl.umn.edu/98190.

9. *Annals* editors-in-chief include Bernard A. Galler, 1979–1987; J.A.N. Lee, 1987–1995; Michael R. Williams, 1996–2000; Tim Bergin, 2000–2003; David A. Grier, 2004–2007; and Jeffrey R. Yost, 2008–2011.

created by historian of science Roger Stuewer at the University of Minnesota.[10] In subsequent years, under the direction of Arthur Norberg, CBI took a keen interest in archival records created by ACM's officers and luminaries, including such figures as Edmund Berkeley, Alan Perlis, Charlie Bachman, and others, as well as the notable HOPL conferences, and in 2008 became the public repository of the ACM headquarters archives.[11] This aspect of ACM's history is consequential for this present volume, since access to archival records creates an essential precondition for historical scholarship, and many of this book's chapters draw on these publicly accessible ACM records.

The 1980s was a notable decade for the history of computing, even beyond the early years of *Annals of the History of Computing* and the CBI. CBI was one of several so-called disciplinary history centers, which at the time formed something of a "cohort" and a real-time experiment in connecting academic history to the concerns of professionals and practitioners. Compared with AFIPS, ACM played a rather modest and somewhat distant role in this development. Within a few short years four such disciplinary history centers were launched. In addition to CBI these included centers formed by the IEEE, parent of the IEEE Computer Society; by the American Institute of Physics; and by the American Chemical Society in cooperation with the American Institute of Chemical Engineers. Partnerships with prominent research universities emerged as a common attribute of these centers: these were, respectively, Minnesota, Rutgers, Maryland, and Pennsylvania. NASA had created a NASA history office a year after its founding in 1958, and it too expanded in the 1980s "to support a much wider range of scholarly activities than at any time in the past."[12] All these efforts included archival collecting, oral history programs, and a varied mix of research and commemorative activities.

10. For the founding and early years of CBI, see *CBI Newsletter* articles by A. Norberg. Fall 2003. Twenty-Five Years of the Charles Babbage Institute. *CBI Newsletter* 26(1). Available at www.cbi.umn.edu/about/nsl/v26n1graphics.pdf#page=3; and T. Misa. Fall 2012 and Spring 2013. Charles Babbage Institute: Glimpses of the founding. *CBI Newslett.*, 34(2) and 35(1). Available at www.cbi.umn.edu/about/nsl/v34n2.pdf#page=4 and www.cbi.umn.edu/about/nsl/v35n1.pdf#page=19.

11. See CBI holdings of ACM headquarters records at http://purl.umn.edu/51982; other ACM units at http://tinyurl.com/jran22g; ACM members at http://tinyurl.com/gvx9eq3; and C. Bachman papers at http://purl.umn.edu/40732. See also "ACM Research Materials" at http://history.acm.org/content.php?do=links as well as T. Haigh, E. Kaplan, and C. Seib. 2007. Sources for ACM history. *Commun. ACM*, 50(5): 36–41. DOI: 10.1145/1230819.1230836.

12. R. D. Launius. 1999. NASA history and the challenge of keeping the contemporary past. *Pub. Hist.*, 21(3): 63–81, quote p. 69.

For the history of computing, the 1980s was a notable decade also in the museum world. Work was underway at the National Museum of American History on the landmark exhibit "Information Age: People, Information and Technology," which opened in May 1990 and led to a series of Smithsonian interviews with such notable pioneers as Seymour Cray, Bill Gates, and Steve Jobs. In 1984 the Computer Museum in Boston opened its doors to the public in a high-profile location on the Boston waterfront. Supported originally by the Digital Equipment Corporation, one of the mainstays of Boston's high-technology Route 128, the Computer Museum felt the nation's shifting fortunes as the Boston area's minicomputer industry waned and as California's Silicon Valley grew in economic importance. In 1999–2000, the Computer Museum in Boston was closed and most of its collection shipped to NASA Ames's Moffett Field in California. In 2002, the newly organized Computer History Museum acquired its present location in nearby Mountain View and prepared the way for its main public exhibit, "Revolution: The First 2000 Years of Computing," which opened to great acclaim in 2011.

In 2004, at the prompting of historically minded members David Wise and Rick Snodgrass, the ACM Council created a History Committee and charged it with "foster[ing] preservation and interpretation of the history of the ACM and its role in the development of computing." After a period of organizational work bringing together ACM members with professional historians from the academic and museum worlds, the History Committee embarked on commissioning a set of oral histories with ACM officers and prize winners. Greater visibility of the A. M. Turing Award prompted a multi-year project to expand the prize's website as a means to garner visibility for the awardees and expand recognition of the ACM's role in computing, an effort that has been further expanded with the commissioning of a comprehensive set of oral histories with all living Turing awardees. Another committee activity that directly shaped this present volume is an annual program of fellowships to facilitate historical research on ACM's history. Nearly two dozen fellowships so far have supported a wide variety of research projects done by computer scientists, educators, and academic historians. The results include five completed doctoral dissertations, two published books, and numerous conference presentations, journal articles, and, not least, the chapters in this book. Two hands-on workshops organized by the committee—on archival principles and practices (2014) and on oral-history methods (2016)—have connected ACM members and other computing practitioners with up-to-date professional practices in history. A third research-oriented workshop, held in conjunction with the annual meeting of the Society for the History of Technology (2015) and in cooperation with its special interest group in computing history, or SIGCIS, was an opportunity to present, discuss, and refine nine of the chapters in this book.[13]

1.3 Chapters of ACM History

This book deals with ACM's organizational history, with its historic role in defining "computer science" in the 1960s and its evolution in subsequent decades, with some of its notable special interest groups, and with several prominent research activities. Many of the chapters deal with "science and society" issues, including the debates about how ACM should deal with controversial social and political issues. It is not, however, a formal history of ACM viewed from a unified perspective.[14] Instead, readers will find here a set of chapters that tells the story of intriguing aspects of computing history—where, for each, the ACM played a prominent role. The perspective is a mosaic rather than a linear timeline; the book presents compelling episodes rather than a single cohesive narrative.

The book's first theme, "Defining the Discipline," examines the 1960s and 1970s as the field of computer science was taking something like its presently recognizable form. At the time, computing projects in government, industry, and academia were conducted by people with multiple and distinct disciplinary identities, including mathematics, engineering, physical science, library science, and data processing. Appropriately enough, Janet Abbate's chapter 2 situates the emergence of computing as a proper discipline within the National Science Foundation. The problem was that computing figured in *multiple* NSF programs, including the agency's funding and support for specific computing projects in varied scientific fields, for general campus computing resources, and for mathematical research. Abbate follows the engagement in these debates of ACM presidents Alan Perlis, George Forsythe, and Anthony Oettinger, as NSF evolved initially its dedicated Office of Computing Activities by 1967 and, eventually, the full-scale directorate

13. M. Hall. 2012. Understanding ACM's past. *Commun. ACM* 55(12): 5. DOI: 10.1145/2380656.2380657; and T. J. Misa. 2015. Computing is history. *Commun. ACM*, 58(10): 35–37. DOI: 10.1145/2814845.

14. Formal histories of several of the engineering societies include B. Sinclair. 1980. *A Centennial History of the American Society of Mechanical Engineers, 1880–1980*. University of Toronto Press, Toronto; T. S. Reynolds. 1983. *75 Years of Progress: A History of the American Institute of Chemical Engineers, 1908–1983*. American Institute of Chemical Engineers, New York; A. M. McMahon. 1984. *The Making of a Profession: A Century of Electrical Engineering in America*. Institute of Electrical and Electronics Engineers, New York.

for Computer and Information Science and Engineering (CISE). She also assesses ACM's 1968 model curriculum for computer science.

It is worth recalling that the various terms in use, including computer sciences, information sciences, data processing, and the singular "computer science," were diversely defined, and that a lively debate ensued about whether computing was best considered to be a useful scientific tool, a bounded scientific domain, or a set of novel phenomena calling out for scientific understanding. "Computer science is the study of computers . . . the computer is not just an instrument but a phenomenon as well, requiring description and explanation," asserted three Carnegie Institute of Technology luminaries—Allen Newell, Alan Perlis, Herbert Simon—while Edsger Dijkstra, a friendly rival, countered that "Computer science is no more about computers than astronomy is about telescopes." Donald Knuth, author of the monumental and multi-volume *Art of Computer Programming* (1968 et seq.), offered a definition that explicitly aligned with his Carnegie colleagues: "computer science is the study of algorithms . . . one of the 'phenomenon surrounding computers'."[15] This spirited discussion, one might observe, involved *five* current or future Turing awardees (and with Simon also a Nobel Prize).

For its part, the ACM "Curriculum 68" report pragmatically defined computer science not with any single essence—formal theory, computer programming, and algorithms were possible contenders[16]—but rather by describing three dimensions of the emerging field. "Information structures and processes" centered on data structures, programming languages, and models of computation. "Information processing systems" pointed to computer design and organization, translators and interpreters, and computer operating systems. And "methodologies" enumerated applications such as numerical mathematics, symbol manipulation, data processing and file management, graphics, simulation, information retrieval, and artificial intelligence that featured common structures, processes, and techniques.[17] Text-

15. A. Newell, A. J. Perlis, and H. A. Simon. 1967. Letters: computer science. *Science*, 157(22): 1373–1374 at http://www.jstor.org/stable/1723308; P. J. Denning and C. H. Martell. 2015. *Great Principles of Computing*, quote p. 1 (Edsger Dijkstra). MIT Press, Cambridge, MA; D. E. Knuth. 1974. Computer science and its relation to mathematics. *The American Mathematical Monthly*, 81:323–343, on 324.

16. On the decades-long quest for the "essence of computing as a discipline," see M. Tedre. 2015. *The Science of Computing: Shaping a Discipline*, quote p. 5. CRC Press, Boca Raton.

17. W. F. Atchison, et al. 1968. Curriculum 68: Recommendations for academic programs in computer science: A report of the ACM Curriculum Committee on computer science. *Commun. ACM*, 11(3): 151–197, quotes on 154–155. DOI: 10.1145/362929.362976.

books and encyclopedias largely followed this inclusive and enumerative definition of the field.[18]

With attention to these national policy discussions, including high-profile reports by the National Academy of Science and other blue-ribbon bodies, Abbate narrates the emergence of computing as a legitimate scientific discipline, worthy of research funding and justified on its own terms. Her discussion is essential background for understanding ACM's role in defining computer science and its continuing role in shaping computer science education. It also serves as a balance to historical accounts focusing on the prominent role of the Defense Department's legendary Information Processing Techniques Office in supporting computer graphics, time sharing, artificial intelligence, and networking.[19]

Joseph November's chapter 3 complements and extends Abbate's. Forsythe was the founder of computer science at Stanford University (1965) as well as ACM president during the years 1964–66. "He, more than any other man, is responsible for the rapid development of computer science in the world's colleges and universities," according to Donald Knuth. November details Forsythe's shifting discipline- and institution-building strategies, as he himself made a career transition from mathematics and meteorology to computing and computer science. As late as 1960, a textbook he co-authored on differential equations envisioned computers as a tool for mathematics rather than an object of scientific study in its own right. His participation the following year in a new "Division of Computer Science" within Stanford's math department raised the question of how to justify and consolidate the fledging endeavor, especially since some influential Stanford faculty thought computing was too much like "plumbing" and lacked respectability as an intellectual field. Stanford's "Computer Science Department" was formally created in 1965 by the legendary Fred Terman, hailed as the founder of Silicon Valley, as a means to expand computing and re-orient the university toward externally funded research. Forsythe was its founding chair. It's odd to hear today, but Stanford faced obstacles to hiring John McCarthy and Edward Feigenbaum into the mathematics department, and so the new department became a free-standing entity in the School of Humanities and Sciences (it moved to the engineering school in 1985). Forsythe's early

18. See S. V. Pollack, editor. 1982. *Studies in Computer Science*, pp. 31, 35–48. Mathematical Association of America, Washington, DC; A. Ralston and C. L. Meek, editors. 1976. *Encyclopedia of Computer Science*, s.v. "computer science," pp. 314–318. Petrocelli/Charter, New York.

19. A. L. Norberg and J. E. O'Neill. 1996. *Transforming Computer Technology: Information Processing for the Pentagon, 1962–1986.* Johns Hopkins University Press, Baltimore.

and unexpected death in 1972, after the first part of his vision was clearly achieved, in November's view hampered achievement of his second and wider vision, which was computer science as a thoroughly interdisciplinary and university-spanning activity, connected to many existing departments and research centers.

Forsythe's daughter, an anthropologist, observed that computer science at Stanford during these years was distinctly inhospitable to women. Irina Nikivincze's chapter 4 picks up exactly this theme through a collective biography of American women who received their doctoral degrees in computer science between 1970 and 1976, years when women received around 7% of the new field's doctoral degrees (and around 15% of master's degrees and a slightly greater proportion of undergraduate degrees). Nikivincze sets computer science into the wider context of women's participation in other scientific fields. She has identified 30 women receiving Ph.D. degrees in computer science from 1970–1976 and subsequently pursuing academic careers (an additional 60 women doctorates went into industry or pursued other careers). Of the 30, she personally did interviews with 7 (3 additional women had been previously interviewed), providing detailed information about a sample of women at an early and formative period of computer science. She found them to be "extraordinary group driven by their passion for the subject and the desire to succeed." The chapter reports on their diverse experiences as they negotiated graduate school and early careers, developed lines of research, struggled for professional recognition, and (for a number) raised families in the midst of busy lives. Membership in, and activities with, ACM and other professional societies are a prominent feature. Her chapter's empirical findings complement a recent book by Abbate on women in computing as well as interviews conducted by the Charles Babbage Institute on women in the computing industry.[20]

ACM played a central role in defining "computer science" as an educational activity; sometimes the field's emergence is even dated to the ACM model curriculum issued in 1968. Sebastian Dziallas and Sally Fincher's chapter 5, "The History and Purpose of Computing Curricula (1960s to 2000s)," makes clear that ACM's curriculum-shaping activities continued far beyond the 1960s and helps in understanding the transformations of computing education during these decades. Their chapter examines the ten major reports issued by ACM and, beginning in 1983–84, in parallel with and then jointly with the IEEE Computer Society. While providing

20. J. Abbate. 2002. *Recoding Gender: Women's Changing Participation in Computing*. MIT Press, Cambridge, MA; existing interviews with women by the Charles Babbage Institute (available at: http://tinyurl.com/jufd2zs) and a further set of 40 CBI interviews supported by the Alfred P. Sloan Foundation.

an excellent overview of the evolution of computer-science education, their chapter is the first in the book to consider the inner workings of ACM—especially the need to coordinate the legion of volunteer committee members as they go about the painstaking business of investigating, developing, revising, and issuing these much-anticipated curriculum reports. Of course model curricula do not evolve in a vacuum, and their chapter considers the recurrent questions about accreditation, including shifting relationships with the Accreditation Board for Engineering and Technology, or ABET, the engineering-oriented accreditation body. To balance the natural visibility of Stanford, MIT, Carnegie Mellon, Purdue, and other research-oriented programs, they devote attention to the ACM's Special Interest Group on Computer Science Education (SIGCSE) and its Committee on Computing Education in Liberal Arts Colleges, where many computer-science faculty teach and where many students take undergraduate degrees.

The book's second organizing theme, "Broadening the Profession," looks outward into the wider society as ACM members and the organization itself engaged with social and political issues—and struggled with balancing a focus on scientific issues with an engagement with the wider world. ACM members did not always agree on the merits or importance of each issue, of course; but it is notable that a range of viewpoints was actively considered, especially compared with other professional engineering societies. In several instances, notably women in the computing profession and extending computing into economically disadvantaged communities, the ACM stands apart from its sister engineering societies.[21] Janet Toland's chapter 6 examines the ACM's efforts in the turbulent years of the 1960s and 1970s to retain its scientific stature (and tax-free status) with some members' desires to have ACM more squarely confront the Vietnam war, the women's movement, and the human rights of Soviet computer scientists. Toland traces the emergence and internal working of ACM's Special Interest Group on Computers and Society, or SIGCAS, seemingly designed to discuss and engage social issues involving computing, and describes its relations with the ACM leadership and the society's public policy committee. ACM founder Edmund Berkeley and ACM member Daniel McCracken each lobbied strenuously for ACM's adopting an overly political stance about the use of computing in warfare. Peter Denning, Jean Sammet, and other ACM leaders at the time aimed to keep the ACM focused on scientific and technical matters

21. Compare with A. M. McMahon. 1984. *The Making of a Profession: A Century of Electrical Engineering in America*, pp. 253–263. Institute of Electrical and Electronics Engineers, New York; and M. Wisnioski. 2012. *Engineers for Change: Competing Visions of Technology in 1960s America*, pp. 111–121. MIT Press, Cambridge, MA.

rather than political issues. Drawing on many personal papers with behind-the-scenes correspondence, Toland's account is full of colorful dialog and delightful detail. By and large ACM engaged with "political issues" only where there was a clear connection to computing, for example with computer-related privacy issues. Her account neatly frames later chapters in this section dealing with other questions of social and political importance.

The women's movement resonated deeply, if unevenly, across the ACM membership as the internal and national debate about the Equal Rights Amendment certainly indicates. Amy Sue Bix's chapter 7 adds an institutional and comparative perspective. Bix, like Nikivincze, sets her account within a rich history of women in science and engineering. Her chapter presents an extended comparison of the Association for Women in Computing (AWC) with ACM's Committee on Women in Computing (ACM-W). In examining the AWC's activities in the late 1970s and 1980s, Bix finds that the fledgling organization, independent from other professional bodies, struggled to focus its attention and resources. Beginning in the 1990s, ACM-W was an informal lobby within ACM for women's issues and in 2006 it was elevated to full Council status; it benefitted mightily from the deep resources of the ACM and the institutional clout that came through its auspices. She indicates a contrast between "AWC's informal enthusiasm sparked by members' personal commitments" compared with "the ACM-W's more institutionally embedded momentum." While pointing to a host of successful efforts in education, conferences, outreach, and institutional mobilization—not least is the mammoth annual Grace Hopper Celebration of Women in Computing—Bix remains aware of signal shortcomings. Although the computer-engineer themed "Barbie" doll initially seemed a positive model for girls, the accompanying 2013 book, *I Can Be a Computer Engineer*, offered a sorry lesson in tone-deaf sexism. With gender equity in computing remaining elusive, it is clear that much remains to be done; Bix's chapter is an essential primer for comprehending the past decades' varied lessons in advocacy efforts for women in computing.

The ACM and the international world are the subjects of the next two chapters, each demonstrating that the ACM was stretched by as well as enriched through the extension of professional relationships to countries beyond the United States. Scott Campbell's chapter 8 deals with an impossibly close neighbor. It provides a capsule history of the several varieties of computer professionalism in both Canada and the United States, dealing with the rise and fall of different professional organizations—some more scientific in their bent, others more attuned to practical bread-and-butter issues. ACM was quick off the mark in Canada, holding its first-ever international meeting in Toronto in 1952. In the 1950s both ACM and the DPMA, noted above, tried to drum up attractive activities, organizational structure, and membership appropriate for a professional organization. In the 1960s these two organizations contended over accreditation in Canada, which is intimately linked to education and the workforce. Canadian Kelly Gotlieb served as editor-in-chief of *Communications of the ACM* during 1962–1965 and of *Journal of the ACM* during 1966–1968, while many Canadians worked closely with the international minded IFIP. Yet attractive programs are not the whole story. Specifically *Canadian* organizations also sprouted up in the Computing and Data Processing Society of Canada, later evolving into the Canadian Information Processing Society or CIPS. CIPS proclaimed itself as the national "industry spokesman," which as Campbell notes is a key attribute of professionalization. The "Canadian Problem," expressed through conflicts about postage stamps and publications and national identity, led Canadians ultimately to split from the ACM-rival DPMA and found their own national "data processing" society. It so happened, all the same, that accreditation standards north and south of the U.S.-Canada border converged neatly. Toronto was and remains one of the larger ACM chapters in North America.

IFIP appears also as a key institutional actor in Ksenia Tatarchenko's chapter 9. It is built around meticulous study of a single episode: the visit of noted Soviet computer scientist Andrei Ershov to the United States in 1965, at the height of the Cold War. Ershov intended to attend the IFIP congress in New York, but these plans were derailed by a delay in his U.S. visa. He did have notable visits with (George Forsythe's) department of computer science at Stanford University and another lecture to the Los Angeles chapter of ACM, in the very heart of the government-dominated aerospace industry. As a counterweight to the traditional image of the Cold War, where implacable opposition reigned between the two superpowers, Tatarchenko provides a striking image of Ershov—chalk in hand (Figure 9.1)—giving a lecture at RAND, the archetypical Cold War think tank! Tatarchenko's archival sources make abundantly clear that deeply personal connections and intense intellectual relationships crossed the Cold War political divide. "It is really true that you are the Novosibirsk branch of the ACM," observed Edward Feigenbaum. Her chapter suggests that these international ties were crucial means for advancing professionalism: demonstrating that one had esteemed colleagues on the other side of the "iron curtain" was a means for showing that the young profession really mattered. Ershov came to the U.S. while Feigenbaum, Forsythe, and John McCarthy, among others, made reciprocal visits to the Soviet Union. After his visit to Stanford, Ershov in a letter related to Forsythe his distinct pleasure in the "singular commonality of our problems and research interests." Her case study joins other recent work showing that computing was one of the bridges in the Cold War, yet to be fully acknowledged.[22]

Arvid Nelsen's chapter 10 forms a neat complement with Toland's chapter 6 on social issues. He investigates a set of practical community-oriented educational efforts designed to realize the promise of computing as a lever of advancement for society. Like Toland's account, Nelsen deals with ACM members advocating awareness of and engagement with these non-college educational programs. While it was relatively easy for Stanford to adopt an elite stance (prioritizing externally funded research over undergraduate education, for example), it proved a great challenge to bring computing education to "disadvantaged" communities in poor neighborhoods. The archival papers of Berkeley and other ACM members help Nelsen piece together an elusive yet gripping story. His institutional focus is the ACM Special Interest Committee on the Social Implications of Computing, or SIC^2 (which traces its antecedents to a Washington, DC group organized in the mid 1960s around the automation question) and the short-lived Committee on Computing and the Disadvantaged, active in the late 1960s and early 1970s. The 18 local educational-training programs that Nelsen identifies were located across the U.S., with concentrations on the East coast and in California. Even with its wariness about being overly political, ACM did devote attention to these efforts with prominent articles appearing in *Communications of the ACM* (its "ACM News" sections are a prime source) and attention by local ACM chapters as well as the special interest group on Computer Personnel Research, or SIGCPR. Ultimately, however, the ACM Committee on Computing and the Disadvantaged, despite organizing an impressive number of activities, lost support and funding (around 1974), and the ACM lost a mechanism to remain connected to these educational and outreach efforts. Like Bix's chapter 7 on women, Nelsen's on educational outreach can serve as a cautionary tale and primer for today's enthusiasts of efforts for computing for all.

The book's third section, "Expanding Research Frontiers," profiles three areas of research activity where ACM members and ACM itself shaped notable advances in computing. Jacob Gaboury's chapter 11 forms an intellectual prequel to the notable activities of ACM SIGGRAPH. His story follows the notable career of David C. Evans, who founded the University of Utah's storied computer science department

22. S. Donig. 2010. Appropriating American technology in the 1960s: Cold War politics and the GDR computer industry. *IEEE Ann. Hist. Comput.*, 32(2): 32–45; K. Tatarchenko. 2010. Cold War origins of the International Federation for Information Processing. *IEEE Ann. Hist. Comput.*, 32(2): 46–57; N. Lewis. 2016. Peering through the curtain: Soviet computing through the eyes of Western experts. *IEEE Ann. Hist. Comput.*, 38(1): 34–47.

and shaped it into a world-leading center for early computer graphics. While at the Bendix Corporation in southern California during the 1950s, one of his colleagues was Harry Huskey who had worked with Alan Turing on the Pilot ACE project. Evans, after a brief stint at University of California–Berkeley, returned to his native Utah in 1965–1966 to found a computer science division within the university's School of Engineering. Funding for a nascent research program in "Graphical Man/Machine Communication" was arranged by IPTO's Ivan Sutherland who, after a brief time at Harvard, also came to the University of Utah. Evans, with a brand-new department, was free to focus both research and education on computer graphics; students chose well-defined "problems" in computer graphics, received top-level support in addressing them, and often enough did research that had field-creating results. Among the Utah graphics alumni are such figures as Ed Catmull, Nolan Bushnell, Alan Kay, John Warnock, and others well-known to graphics experts. Evans and Sutherland formed a company that helped create the computer graphics industry. Gaboury's focus is on Utah and yet he acknowledges the "professionalization of computer graphics through the ACM and its SIGGRAPH special interest group." After the notable 1973 SIGGRAPH conference connected academic research more closely to industry, computer graphics research spread outward from Utah—initially to Xerox PARC, NASA's Jet Propulsion Laboratory, the New York Institute for Technology—often through the movement of Utah graduates.

Rebecca Slayton's chapter 12 takes an up-close focus on ACM-specific activities, including conferences and publications, in creating a new sub-field of computing. Computer security was a concern of at least *six* different ACM SIGs which each pursued its own framing or conceptualization of computer security, leading, she suggests, to an early balkanization of the field. Privacy was a leading concern of SIGCAS (Computers and Society) as Janet Toland's chapter 6 in this volume also documents. Protection of computer resources was a focus of researchers in SIGOPS (Operating Systems). Databases were a joint concern for SIGMOD, SIGIR, and SIGBDP (respectively, Management of Data, Information Retrieval, and Business Data Processing). In the early 1980s SIGSAC (Security, Audit, and Control) emphasized managerial solutions to the problem of computer crime. Some integration occurred in the 1980s when networking and privacy debates encouraged a search for commonality in security framings. The IEEE sponsored the annual Symposium on Security and Privacy, beginning in the 1980s, while the ACM sponsored the Computers, Freedom, and Privacy conferences beginning in the 1990s. ACM members also played important roles in the wide-ranging discussion of "critical infrastructures." In all, computer security (to slightly expand a quote) "appears to be a very hard problem to solve."

Inna Kouper's chapter 13 explores how ACM members responded to changing research frontiers and emerging technologies. Her chapter is focused on a twin transformation: on the technical level, research focused on hypertext to research on the World Wide Web, which was one instantiation of hypertext; and on the organizational level, on the evolution of SIGLINK to SIGWEB, signally a shift to fully embrace the Web and to expansively conceptualize hypertext. Kouper begins with an intellectual history of "hypertext" as it emerged in the visions of Vannevar Bush, Douglas Engelbart, and Theodore Nelson. ACM took a prominent role in sponsoring the landmark *Hypertext '87* workshop at Chapel Hill, North Carolina. At the time, compelling instances of hypertext—"Xerox's NoteCards, Owl's Guide, and especially Apple's HyperCard" as keynoter Andries van Dam put it—existed in a distant, parallel universe with internetworking. Hypertext standards were developed by the Dexter group which authored a three-layer model. Hypertext was at the time a coherent field of research. Notably, a paper proposal by none other than Tim Berners-Lee to the 1991 hypertext conference was outright *rejected*: "as being weak on research and . . . not [containing] references to the relevant work in the field." Rapid growth in Berners-Lee's World Wide Web in the 1990s coexisted with SIGLINK's persisting "vision of hypertext research" of "standalone systems that allowed [users] to explore architectural, aesthetic, and design complexities" as well as exciting work in non-web-based "digital libraries." Eventually SIGLINK, focused on the hypertext domain, broadened its agenda in 1998 to become SIGWEB. Alas, the WWW conferences organized by Berners-Lee's organization soaked up much of the energy and resources of the burgeoning web community, even leading to the odd misperception among some that the WWW was "the first hypertext." Kouper's close study of papers, panels, awards, and conference themes, played out alongside the institutional transformation of an ACM SIG, provides a rewarding perspective on the history of a vital aspect of contemporary computing.

This book is the first to explore the history of ACM as a community of computing professionals. In several ways, it reflects the state-of-the-art in historical research on ACM. Historians are exploring the rich archival records that ACM has itself made available to researchers and supplementing these traditional sources with oral histories of ACM awardees, officers, and members. A great deal of ACM's activities takes place not at its New York headquarters but in the decentralized work of its active special interest groups and local chapters. The ACM History Committee encourages ACM SIGs and makes widely available (through workshops and outreach) the full range of professional practices and standards in history. Twenty SIGs have significant web-, wiki-, or conference-based historical activity.[23]

Notable recent activities include SIGOPS's SOSP History Day (2015), SIGCSE's ongoing Computing Educators Oral History Project, and the roundup of the ACM History Committee activities.[24]

For readers searching for additional information on the "big picture" of ACM there are several places to turn to for further exploration. *Communications of the ACM* has published a handsome number of articles on ACM history, as the collected bibliography at the end of this volume amply attests. Bernadette Longo's biography, *Edmund Berkeley and the Social Responsibility of Computer Professionals*,[25] profiles one of the organization's founders and one of its most colorful and opinionated members. And there is now a full set of online profiles of the winners of the A. M. Turing Prize as well as a full-scale oral history project seeking to create professionally done interviews with all living awardees.[26] It so happens that *29* Turing laureates figure someplace in this present volume—listed here in order of receiving the award: Alan Perlis, Maurice Wilkes, Richard Hamming, Marvin Minsky, John McCarthy, Edsger Dijkstra, Charlie Bachman, Donald Knuth, Herb Simon, Allen Newell, John Backus, Edgar Codd, Dennis Ritchie, Ken Thompson, Niklaus Wirth, Ivan Sutherland, Fernando Corbató, Butler Lampson, Juris Hartmanis, Edward Feigenbaum, Douglas Engelbart, Fred Brooks, Alan Kay, Peter Naur, Frances Allen, Barbara Liskov, Shafi Goldwasser, and, most recently, Whitfield Diffie and Martin Hellman. In addition, modern computing depends implicitly on many others, such as Vinton Cerf and Robert Kahn (2004), for the fundamental internet concept as well as the three eponymous winners for the RSA algorithm (2002).

Research in history, as in other fields, sometimes generates tidy answers but just as often leads to further questions to be asked and problems to be addressed. In assembling this volume, the authors have made significant progress in assembling chapters in ACM's storied history. But of course there is much more to be done.

23. See "SIG History Activities" at http://history.acm.org/content.php?do=sighistory.

24. See "SOSP History Day," October 4, 2015 (Monterey, CA) at http://sigops.org/sosp/sosp15/history; Computing Educators Oral History Project at http://ceohp.org; and the ACM History Committee website and blog at http://history.acm.org and http://history.acm.org/content.php?do=blog.

25. B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals.* Association for Computing Machinery and Morgan & Claypool, New York.

26. See http://amturing.acm.org.

Figure 1.5 **ACM bumper sticker proclaimed "Computing the Future" [n.d.].** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

ACM's leading efforts in defining "computer science" as a field of research and as a respected academic discipline clearly merits additional investigations. While this book outlines the important connections between ACM and Stanford, there remains complementary investigations of other pioneering computer science departments such as Purdue's and Carnegie Mellon's (and elsewhere) where ACM leaders were notably prominent and remain so through today.[27] ACM's special interest groups, or SIGs, have played impressive roles in cultivating the fields of programming languages, graphics, operating systems, computer-science education, and security (among many fields); numerous case studies need to be done to understand the creation, organization, operations, and consequences of the notable conference series that they've run. ACM's role in the wider world of politics, society, and economics also merits further consideration. If computing is creating the future—as the ACM once suggested with an informal bumper sticker (see Figure 1.5) and as Turing laureate Juris Hartmanis emphasized in an acclaimed 1992 National Research Council study[28]—it's a vital matter that a diverse range of professionals, educators, students, and citizens are engaged with its promise. We hope that future volumes in ACM history will take up some of these challenges and opportunities.

27. R. L. Pyle. 2015. *First in the Field: Breaking Ground in Computer Science at Purdue University.* Purdue University Press, West Lafayette, IN; H. Crowther-Heyck. 2005. *Herbert A. Simon: The Bounds of Reason in Modern America.* Johns Hopkins University Press, Baltimore. Compare P. Mounier-Kuhn. 2012. Computer science in French universities: Early entrants and latecomers. *Inform. Cul. J. Hist.*, 47(4): 414–456; R. J. Carnota. 2015. The beginning of computer science in Argentina and the Calculus Institute, 1957–1970. *IEEE Ann. Hist. Comput.*, 37(4): 40–52.

28. J. Hartmanis and H. Lin, editors. 1992. *Computing the Future: A Broader Agenda for Computer Science and Engineering.* National Academy Press, Washington, DC

2

From Handmaiden to "Proper Intellectual Discipline": Creating a Scientific Identity for Computer Science in 1960s America

Janet Abbate

2.1 Introduction

In September 1966, ACM—a major professional society for computer scientists—published a letter to the membership from its new president, Anthony G. Oettinger. Oettinger lamented that U.S. government funding for computing was targeted to serve other disciplines, rather than computer scientists themselves. ACM members, he wrote, "have expressed deep concern over the fact that while very large amounts of money go into computation these days this does not in itself furnish money *for research and education in computer science*. They point out that most of the support for computation is now channeled to computation centers for their service activity." Having brought these concerns to the National Academy of Science and the National Science Foundation, Oettinger had found "numerous misconceptions about computer science firmly expressed within high councils of science and government." At the top of his list of misconceptions was, "The computer is just a tool and not a proper intellectual discipline."[1]

Indeed, computer science as an academic field was still quite new in 1966—but it was growing fast. Electronic digital computers had been invented during World War II, and the postwar years saw escalating demand for computers in science, industry, and government.[2] A number of universities began offering computer science degrees in the early 1960s, and by 1963 at least 28 universities were offering bachelor's, master's, or doctoral programs in computer science.[3] Professional societies complemented these efforts to create a scientific infrastructure for computer science. In 1947, an informal group of computer experts formed the first professional society devoted to computing, the Association for Computing Machinery (ACM). As Edmund C. Berkeley, one of the chief founders of ACM, saw it, the new society would "advance the science, design, construction, and application of the new machinery for computing."[4] Other computer groups were formed within the American Institute of Electrical Engineers (1946) and Institute of Radio Engineers (1951); when these two societies merged in 1963 to form the IEEE, their computer groups were combined and eventually in 1970 they became the IEEE Computer Society. Professional societies supported scientific work by providing a forum for research results through academic conferences and peer-reviewed journals, by assisting in curriculum development, and—as we will see—by raising a collective voice for computer scientists in policy matters.[5]

1. A. G. Oettinger. 1966. President's letter to the ACM membership. *Commun. ACM*, 9(12): 838, emphasis in original. Since ACM had no direct representation in NAS or NSF, Oettinger depended on the goodwill of Harvard physicist Harvey Brooks to act as an intermediary. "By unanimous vote of the [ACM] Council at its meeting of August 29th, I was instructed to forward a letter expressing these concerns to the Committee on Science and Public Policy of the National Academy of Sciences (COSPUP). Dr. Harvey Brooks, Chairman of COSPUP, very kindly initiated discussion of our problem both within the National Academy of Sciences and within the National Science Board of the National Science Foundation."

2. For overviews see P. E. Ceruzzi. 1998. *A History of Modern Computing*. MIT Press, Cambridge, MA; M. Campbell-Kelly and W. Aspray. 1996. *Computer: A History of the Information Machine*. Basic Books, New York.

3. W. F. Atchison and J. W. Hamblen. 1964. Status of computer sciences curricula in colleges and universities. *Commun. ACM*, 7(4): 226–227.

4. Quoted in B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals*, p. 80. Association for Computing Machinery and Morgan & Claypool Publishers, New York.

5. See chapter 5 in this volume by Dziallas and Fincher for the role of ACM and IEEE CS in curriculum development and how these efforts helped define the identity of computer science.

The National Science Foundation, the primary source of government funding for scientific research in America, did not initially recognize computer science as a scientific field on a par with other disciplines.[6] NSF had been founded in 1950 with a mandate to support "basic scientific research," and many of its staff viewed computer science—with its emphasis on building machines—as a mere handmaiden for the "real" sciences.[7] NSF operations were structured largely along disciplinary lines, with units (programs, sections, or divisions) for mathematics, astronomy, physics, chemistry, biology, and engineering sciences.[8] The organizational control wielded by the established disciplines, as well as NSF's emphasis on basic research, put the emerging field of computer science at a disadvantage. In this context, the notion of computing as a "science" and the appropriateness of NSF funding for computing researchers were both contested. Yet fairly quickly, computer scientists were able to change their status at NSF from handmaiden to "proper intellectual discipline." Computing research began to obtain dedicated funding through a series of NSF units, starting with the Office of Computing Activities in 1967 and culminating with the creation of the directorate for Computer and Information Science and Engineering (CISE) in 1986.

This chapter addresses the historical question of how computer scientists were able to gain intellectual respectability and an institutional foothold at NSF in such a short time. The answer reveals much about the politics of scientific identity and

6. While computer scientists also got significant funding from military agencies, that tended to be for mission-oriented projects; NSF provided the majority of funding for basic computer science until 1982; see National Research Council. 1999. *Funding a Revolution: Government Support for Computing Research*, Fig. 3.6. National Academies Press, Washington, DC. (Available at http://wayback.archive-it.org/5902/20160210155450, http://www.nsf.gov/statistics/nsf99346/). Moreover, funding from military agencies such as DARPA did not depend on computer science being seen as a *scientific* discipline. As NSF program manager Frederick W. Weingarten pointed out, "The generals knew that computers were very important. They weren't hung up about whether it was a basic discipline or not" (F. Weingarten. 1990. Interview by William Aspray, Charles Babbage Institute, OH 212, p. 18. Available at http://purl.umn.edu/107706). Thus, in terms of shaping a disciplinary identity, computer scientists' interactions with NSF were more important than their interactions with the military. For DARPA's role in computing, see J. Abbate. 1999. *Inventing the Internet*, Chap. 1. MIT Press, Cambridge, MA; and A. L. Norberg and J. E. O'Neill. 1996. *Transforming Computer Technology: Information Processing for the Pentagon, 1962–1986*. Johns Hopkins University Press, Baltimore.

7. *National Science Foundation Act of 1950*, PL-507, 81st Congress, 2nd Session.

8. Although "engineering sciences" might seem like a natural home for computer science research, computer scientists were not very involved with NSF's engineering division in the 1960s and early 1970s. Aside from a brief flurry of activity in information theory in FY 1961, the Engineering division did not focus on computing topics until it started a program for computer engineering

the multiple meanings of *discipline* in the context of 1960s America. In framing this investigation I assume that "science" is a socially constructed rather than natural category, and that the term "discipline" has had multiple meanings that are historically specific. I draw on Thomas Gieryn's theory of "boundary-work," which holds that one strategy of those who wish to establish a new scientific field is to rhetorically redraw boundaries so that an area formerly considered "outside" of science now appears to be "inside." Such maneuvers are possible because definitions of science change over time and encompass a diverse and sometimes contradictory set of characteristics, allowing actors to selectively emphasize those aspects that best support their own particular struggle. As Gieryn observes, "What science becomes, the borders and territories it assumes, the landmarks that give it meaning depend on exigencies of the moment—who is struggling for credibility, what stakes are at risk in front of which audiences, at what institutional arena?"[9]

In the case of computer science, a range of cultural assumptions about both the nature of science and the nature of computing were available in public discourse of the 1960s. As I have described elsewhere, computer scientists who publicly asserted the scientific status of their field tended to draw on three distinct meanings of science (sometimes in combination).[10] The first line of argument defined science as the study of natural phenomena. While including computing machines in this category may seem counterintuitive, advocates argued that computer scientists actually study *information*, which occurs naturally in the world (in DNA, for example). This framing appears in the ACM's 1965 Preliminary Recommendations for a computer science curriculum, which offered this definition: "Computer science is concerned with *information* in much the same sense that physics is concerned with energy. . . . As physics uses energy transforming devices, computer science uses information transforming devices."[11] A second approach defined science as the derivation of ab-

in 1979 (see NSF Annual Reports for these years). Dian Olson Belanger's history of engineering at NSF argues that the engineers had initially tried to emulate scientists, but by the 1960s had become disillusioned with this strategy and "moved to assert that engineering deserved recognition on its own merits" See D. O. Belanger. 1998. *Enabling American Innovation: Engineering and the National Science Foundation*, p. 75. Purdue University Press, West Lafayette, IN.

9. T. F. Gieryn. 1999. *Cultural Boundaries of Science: Credibility on the Line*, x–xi. University of Chicago, Chicago.

10. J. Abbate. October 2013. Is computer science "science"? A half-century debate. Keynote talk, 2nd International Conference on History and Philosophy of Computing, Paris.

11. ACM Curriculum Committee on Computer Science. 1965. An undergraduate program in computer science: Preliminary recommendations. *Commun. ACM*, 8(9): 543–552, on 544.

stract ideas from concrete phenomena. ACM president George Forsythe provides an apt example in a 1965 letter to the membership, where he posed the rhetorical question, "If the computer is just a tool, why should there be an international society devoted to it?" His answer was that the "essence" of computer science was not machinery but abstract theory. Forsythe wrote, "As with Mathematics or Statistics or Physics, so also with Computer Science: It is more efficient to abstract the essence of the subject from its applications and study the essence itself."[12] A third strand of argument presented the experimental method as the defining characteristic of science. Herbert Simon took this position in his 1969 book *The Sciences of the Artificial*, in which he argued that computer systems had become so complex that their behavior could not always be predicted and must be discovered empirically.[13]

I see no reason to doubt that computer scientists' beliefs about science were sincerely held or that they aligned with practitioners' personal experiences as researchers. But statements about the scientific nature of computing could also be made to serve a more public and strategic purpose. I argue that in the institutional arena of US science policy and for an audience of both computing insiders and skeptical outsiders, computer scientists performed two forms of boundary-work, using mutually reinforcing discursive and institutional strategies. Institutionally, they pursued strategies to include representatives of computer science within bodies that made or influenced science funding decisions. Discursively, they argued in speeches and publications that computer science met what they claimed were the crucial criteria for scientific identity.

Looking at discipline formation as boundary-work reveals an outward-facing meaning of *discipline,* as a claim to status and turf; this sense is apparent in many comments by contemporary actors. But some of the debates over the scientific status of computing were conducted among computer scientists themselves and not aimed at policy makers. What was at stake in these internal discussions? I suggest that the desire for a disciplinary identity had a second meaning for the computer science community: as a search for intellectual coherence and consensus on practice. Computer science was a new field whose definition and scope varied widely between institutions. Indeed, not all computing practitioners agreed that their field was a science at all, some preferring to identify as engineers or business

12. G. E. Forsythe. 1965. President's letter to the ACM membership: Why ACM?" *Commun. ACM*, 8(3): 143–144. As Joseph November describes in this volume, Forsythe saw computer science as natural, experimental, *and* abstract, combining the three definitions of science discussed here.

13. H. A. Simon. 1969. *The Sciences of the Artificial*, p. 21. MIT Press, Cambridge, MA.

professionals. By declaring their field to be a science, computer scientists sought both to erase the boundary separating them from the scientific establishment and to reassure themselves that their field had the potential to become—like other sciences—a "proper" intellectually coherent discipline. Gregory Good describes this aspirational meaning of *discipline* as "a combined social contract and conceptual and practical framework," a promissory note that keeps colleagues loosely aligned as they build their field through a process of "assembly."[14] In Good's model, "Scientists assemble disciplines using many elements: phenomena, methods, instruments, theories, analytical techniques, and institutional tools such as journals, government bureaus, and university positions."[15] These elements can provide a basis for community and collaboration while the discipline's conceptual framework is gradually filled in. Viewing discipline-building as assembly helps explain how the institutional development of computer science in the 1960s (professional societies, journals, university positions, and an NSF office) could precede any firm consensus on the field's content.

This chapter begins with an assessment of the status of computing at NSF at the beginning of the 1960s. I then examine examples of institutional and discursive boundary-work that ACM leaders and other computer scientists performed as a means to improve their standing and resources. Finally, I describe the successful outcome of these efforts, as embodied in the new NSF Office of Computing Activities. The evidence for my historical argument is drawn from a variety of published and archival primary sources. Contemporary policy reports by the National Academy of Science and the President's Science Advisory Committee supply important examples of how public discourse about science became linked to material resources. NSF Annual Reports provide details on program activities as well as the public-facing view of the rationale for these, while the *Organizational Development of the National Science Foundation* is invaluable for tracking year-by-year realignments that shifted with the political winds. ACM publications, notably its flagship *Communications of the ACM*, convey the views and policies of ACM leaders as well as feedback from ordinary ACM members in the form of letters. These contemporary historical sources are complemented by retrospective oral histories from the CBI collection, which provide insight into the motives and perceptions of some key participants.

14. G. A. Good. 2000. The assembly of geophysics: Scientific disciplines as frameworks of consensus. *Stud. Hist. Philoso. Sci. (Part B)*, 31(3): 259–292, on 259, 261.

15. Ibid., 259.

2.2 The Status of Computing at NSF in the Early 1960s

The National Science Foundation made occasional grants for computer science research in the 1950s, but most of its computing funds went to university computer centers to support research in other scientific fields. NSF's earliest efforts to fund university computer centers were overseen by mathematician and computer pioneer John von Neumann. Von Neumann had hands-on experience with university computing, having initiated and overseen the creation of a computer at Princeton's Institute for Advanced Study from 1946–1951. In February 1955 von Neumann was asked to head NSF's ad hoc Advisory Panel on University Computing Facilities, which was able to persuade the National Science Board to include computers in the funding category for large-scale scientific facilities.[16] The first five grants were made in 1956, and in 1959 NSF created a dedicated budget for computer facilities.

Kent Curtis, who took charge of the computer centers program in 1967, recalled that the program helped establish computer science, "because it provided the first experimental equipment which the computer scientists had to use in research in their own field."[17] But the emphasis on funding computer centers as a service function also reinforced the subordinate status of computing at NSF. This is reflected in the NSF annual reports for fiscal years 1959–1962, which mention computers only in terms of facilities, summer programming courses for college professors, "supplementary teaching aids," and "mechanical aids" for information handling.[18] The 1962 annual report reiterates this service role: "Computers are proving of ever-increasing value as *tools* for research and training in virtually every imaginable scientific field"; nowhere does the report mention computer science.[19] Funding for computer science *research* was channeled through NSF's Mathematical Sciences program. This unit had well-established disciplinary norms, and by mathematicians' standards, proposals for computer science projects often looked

16. T. A. Keenan. 1990. Interview by W. Aspray, Charles Babbage Institute, OH 217, p. 7. Available at http://purl.umn.edu/107401; W. Aspray and B. O. Williams. 1994. Arming American scientists: NSF and the provision of scientific computing facilities for universities, 1950–1973. *IEEE Ann. Hist. Comput.*, 16(4): 61. Aspray and Williams note that von Neumann was awarded NSF's first computing grant in 1954, for a conference exploring applications of computers to meteorology.

17. K. K. Curtis. 1987. Interview by J. Minker, pp. 3–4. Available at http://purl.umn.edu/107238.

18. Quotations are from *Annual Report of the National Science Foundation, FY 1960*, pp. 103, 118. NSF, Washington, DC.

19. *Annual Report of the National Science Foundation, FY 1962*, p. 60, emphasis added. NSF, Washington, DC.

less polished than competing mathematical proposals.[20] Thus, computer science was positioned at NSF as either a handmaiden to other sciences (in the facilities program) or a second-class form of mathematics (in the research program).

A series of ACM presidents bemoaned the status quo at NSF in speeches and letters to their membership. Alan J. Perlis, who helped found the computer science department at Carnegie Tech (now Carnegie Mellon), argued in 1963 that dedicated funding for computer science was "urgently needed . . . to develop computing without the constraints now attached to it as a purely derivatory activity."[21] His successor, George Forsythe of Stanford, cynically noted, "I see no practical alternative today to our *assuming the role* of one of the mathematical sciences" as a strategy to obtain federal funding.[22] But as Oettinger pointed out in 1966, it was unlikely that computer science would be fairly represented "in a body dominated by mathematicians, . . . whose outlook on computing, if at all understanding, is from the point of view of a computer *user*."[23] To change the status of computer science at NSF, ACM leaders and other concerned computer scientists would employ both discursive and organizational strategies to position themselves within the mainstream of science.

2.3 Organizational Boundary-work: Getting a Seat at the Table

ACM leaders strove to get their voices heard in the organizations that controlled funding decisions. One strategy was to join and become active in professional organizations at the bottom of the science policy pyramid—where membership was relatively easy to obtain—and use these channels to gain influence at higher levels. Thus, in the spring of 1961, ACM joined the Conference Board of Mathematical Sciences, an umbrella organization that included the American Mathematical Soci-

20. See, e.g., W. R. Adrion, Interview by William Aspray, Charles Babbage Institute, OH 211, p. 12. Available at http://purl.umn.edu/104300; F. Weingarten. 1990. Interview by William Aspray, Charles Babbage Institute, OH 212, p. 9. Available at http://purl.umn.edu/107706. Joseph November makes a similar point in chapter 3 of this volume, where he describes Forsythe's concern that computer science's "priorities were sufficiently different from mathematics that the latter field had no mechanisms to reward or protect those who did meaningful work to develop algorithms and computational techniques" and that computer science would therefore "be viewed by mathematicians as the lowest form of mathematics."

21. A. J. Perlis. 1963. Computation's development critical to our society. *Commun. ACM*, 6(10): 642.

22. G. E. Forsythe. 1966. President's letter to the ACM membership. *Commun. ACM*, 9(4): 244, emphasis added.

23. A. G. Oettinger. 1966. President's letter to the ACM membership. *Commun. ACM*, 9(12): 839, emphasis added.

ety, the Association for Symbolic Logic, the Institute of Mathematical Statistics, the Mathematical Association of America, the National Council of Teachers of Mathematics, and the Society for Industrial and Applied Mathematics.[24] ACM did not join CBMS simply because it was a "natural" intellectual home; indeed, many ACM members did not identify with the "mathematical sciences" and questioned why ACM should pay even a modest membership fee to belong.[25] But one of CBMS's functions was to provide advice to government agencies on mathematical issues, including funding needs. Urging skeptical members to continue support for CBMS in 1965, Forsythe as ACM president warned, "It is not easy for a new discipline to break into the Establishment! . . . Since research and development and education in computing are vastly expensive, there is no alternative but to maintain as good relations as possible with the only organization having the kind of funds needed—the US Government."[26] In this usage, a *discipline* functions as a political unit engaged in a boundary-drawing struggle.

Forsythe argued that ACM's investment of dues money, and its officers' time attending meetings, was beginning to pay off. The National Academy of Sciences' Division of Mathematical Sciences was in the process of forming a Committee on Support of Research in the Mathematical Sciences (COSRIMS) as part of a series of discipline-specific reports on research funding needs requested by the NAS Committee on Science and Public Policy. As Forsythe explained, "Such committees will have an important effect on the allocation of national funds (and hence effort) in the coming decade. It is extremely important to computing and to the nation that our field be adequately represented on the committees. But such representation doesn't occur automatically." He pointed out that COSRIMS members had been chosen on the advice of CBMS, and that ACM's membership in CBMS had shifted the boundary of "mathematical sciences" to the advantage of computer scientists: "It is largely due to ACM's membership [in CBMS] . . . that the Conference Board is beginning to think of 'the mathematical sciences, including computing' instead of just 'mathematics' when important decisions are made."[27] A few months later, Forsythe's successor, Oettinger, reminded ACM members of the political importance of COSRIMS as he asked the society to contribute needed funds for the COSRIMS report. NSF had provided initial support for COSRIMS, but this amount

24. H. D. Huskey. 1961. From the President of ACM. *Commun. ACM*, 4(4): 1.

25. While ACM's allegiance with CBMS would prove politically effective, many computer scientists remained uncomfortable identifying as mathematicians, and ACM eventually left CBMS in 1985.

26. G. E. Forsythe. 1965. Official ACM. *Commun. ACM*, 8(7): 424.

27. Ibid.

was not sufficient to complete and publish the three-volume report, and professional societies were being asked to make up the difference. Oettinger urged ACM to pay this money in order to secure long-term research funding: "I must emphasize that, at this time, COSRIMS is our only major official point of contact with the powerful policy-recommending machinery of the National Academy of Sciences-National Research Council."[28] The ACM Council agreed to contribute $1000, and the report was published in 1968 and well received.[29]

In this example, organizational and discursive boundary-work went hand in hand: by changing their institutional location, computer scientists were able to shift the boundary of "mathematical sciences" to include computer science, and the COSRIMS report helped shift the public identity of computer science from a purely service role to a research field. The COSRIMS report powerfully supported the computer scientists' ambitions, describing investment in computer science as a matter of national urgency. While the report noted that computer science "shares with mathematics the role of handmaiden to all of science and technology," it emphasized that computer science should also be supported as its own research area.[30] Harvey Brooks, chair of the Committee on Science and Public Policy, warned in his letter presenting the report, "The development of computer science only as a by-product of the application of computer techniques in other fields often results in failure to develop a distinctive body of theory and technique in computer science in

28. A. G. Oettinger. 1966. President's letter to the ACM membership. *Commun. ACM*, 9(12): 839.

29. The first two volumes of the report were policy recommendations for research and education in mathematical sciences. The third volume was a collection of essays on contemporary mathematical topics, described by a *New York Times* reviewer as "probably the best book on higher mathematics ever written" (H. Schwartz. 1969. "Review of *The Mathematical Sciences: A Collection of Essays*. Edited by the National Research Council's Committee on Support of Research in the Mathematical Sciences (Cosrims) with the Collaboration of George A. W. Boehm. Cambridge, MIT Press," *New York Times*, April 20, 1969, p. 39). A review of the first two volumes in *Science* observed more cynically that the report "seeks to build a case that will encourage federal agencies to lay on the dollars with a generous hand," but also noted that "preparation of the report provided an unusual opportunity for people from all branches of pure and applied mathematics to learn more of one another's problems and opportunities" and quoted Brooks's call to support computer science "in its own right" (L. J. Carter. 1968. More funds urged for science's "leading wedge". Review of *The Mathematical Sciences: A Collection of Essays*, Cosrims and G. A. W. Boehm, editors. *Science* New Series 162(3856): 883, 885).

30. Committee on Support of Research in the Mathematical Sciences COSRIMS. 1968. The mathematical sciences: A report, p. 94, quoting Newell. National Academy of Sciences, Washington, DC.

its own right."[31] The report recommended that "at the national level special priority be given to support of the expansion of research and graduate study in computer science," citing a "critical shortage of research leaders," and concluded that "every effort should be made to support as many good proposals as possible for research in computer science."[32] ACM leaders' strategy for making themselves heard in the top forums for science policy advice had succeeded.

ACM did not confine itself to joining established policy groups such as CBMS; it also created its own institutional mechanisms for monitoring and influencing government science policy and funding. For example, in November 1966 Oettinger persuaded the normally frugal ACM Executive Committee to hire a Washington correspondent, James P. Titus, to gather intelligence on federal activities and report to the membership in a monthly column in *Communications of the ACM*.[33] The topics covered a range of issues, from proposed computer privacy regulations to guidelines for software patents, but Titus frequently highlighted opportunities for, and obstacles to, government support for computer science.[34]

Another initiative was begun by ACM president Alan J. Perlis, who in April 1964 convened the whimsically named "Commission of Thoughtful Persons" to consider the society's future. The commission, which was evenly split between academic and industry practitioners, was "unanimous in recommending that the ACM adopt the policy that as much government support as is consonant with legal activities be obtained."[35] One strategy the group suggested was to offer government bodies free expert advice on computing issues, a service that would allow computer scientists to cultivate government connections while conveying the concerns of ACM members on policy issues. At the ACM Council meeting the following August, past-president Oettinger reported that he was forming such a group, called the Committee on Government Relations, which would be "a source of information about the professional

31. Ibid., Brooks cover letter, Feb. 2, 1968, pp. iii–iv.

32. Committee on Support of Research in the Mathematical Sciences COSRIMS. 1968. The mathematical sciences: A report, pp. 17, 205–206. National Academy of Sciences, Washington, DC.

33. A. G. Oettinger. 1966. President's letter to the ACM membership. *Commun. ACM*, 9(12): 712.

34. See J. P. Titus. 1967. ARPA: A visible means of support (Washington Commentary). *Commun. ACM*, 10(8): 519–520; and J. P. Titus. 1967. The patchwork nature of government computing support (Washington Commentary). *Commun. ACM*, 10(9): 589–592.

35. G. E. Forsythe. 1964. President's letter to the ACM membership. *Commun. ACM*, 7(8): 508. Besides Perlis the initial members included Paul Armer (RAND), John W. Carr III (University of Pennsylvania), ACM Executive Director Don Madden (IBM), and Walter Ramshaw (United Aircraft). The second meeting also included Dick Hamming of Bell Labs and Forsythe, the new ACM president.

viewpoint of ACM members for various Government agencies."[36] The committee's other three members were equally distinguished and well-placed: ACM President Alan Perlis; Paul Armer, Head of the Computer Science Department at the RAND Corporation; and J. C. R. Licklider, who was concluding a stint as director of ARPA's Information Processing Techniques Office to become a consultant for IBM. The group remained active for several years, and in December 1966 Oettinger reported that as a result of its efforts, "several agencies of the U.S. government have now expressed an interest in direct communication with the computing field." The group was enlarged and renamed the Government Advisory Committee, with a mission "to insure direct and effective participation by the computer science and engineering community in discussions that may affect its future." The committee would not only answer government inquiries but also take the initiative to "speak out freely, forcefully, and intelligently on issues affecting computing and build a bridge between the government and the computer field at large."[37]

As these examples illustrate, ACM's organizational boundary-work created multiple platforms for influencing policy and reconstructing the public image of computer science. What was said in those public forums is the subject of the next section.

2.4 Discursive Boundary-Work: Establishing a Scientific Identity

Computer scientists knew that they had to be recognized *as* scientists in order to be eligible for NSF funding and participation in U.S. science advisory bodies. For many researchers, science also had a normative intellectual status to which they aspired. Asserting their membership in a *scientific* discipline functioned as a claim to intellectual rigor and prestige and provided meaning to their personal goals and experiences as researchers. But since computer science did not fit the canonical model of science—having neither a codified body of theory nor a domain of the natural world associated with it—its advocates were forced to explain and justify their claims to scientific identity. Throughout the 1960s, therefore, computer scientists took every opportunity to assert, both among themselves and to their scientific peers, that they were truly scientists. This discourse reveals both computer scientists' tendency to equate *discipline* with *science* and a striking variety in their criteria for what makes a science.

36. Ibid., 633–636.

37. A. G. Oettinger. 1966. President's letter to the ACM membership. *Commun. ACM*, 9(12): 839. The new committee was headed by former ACM President Harry D. Huskey.

The effort to establish scientific credibility began with the name of their field, which was a subject of ongoing debate. As noted above, NSF had classified computing research under "applied mathematics" rather than assigning it an independent scientific identity. The term "computer sciences" was introduced in 1959 by Louis Fein, a consultant who wrote a widely read report on the state of computing education in U.S. universities.[38] While Fein never defined "science," his concept of "discipline," which was modeled on mathematics, drew on the contemporary understanding of science as a creative, theoretical, problem-driven enterprise. Fein's criteria for a discipline included: "Workers in the field do nonroutine intellectual work," "The field has sometimes been axiomatized," and "The field is open, i.e., problems are self-regenerating."[39] In 1963, Saul Gorn of the University of Pennsylvania used a similarly expansive label in an article promoting "The Computer and Information Sciences: A New Basic Discipline." While Gorn did not define what constituted a scientific discipline, his implied criteria included a unique subject domain—"the synthesis and analysis of mechanical languages and their processors"—and a theoretical dimension, which for him included "probabilistic information theory" as well as "a theory of mechanistic information and its processing."[40] Gorn's article was addressed to an audience of mathematicians, appearing in the *Society for Industrial and Applied Mathematics Review*, and seems aimed at persuading the scientific establishment that computer science was, in fact, a "basic science."

Other early leaders, such as operations research specialist Philip M. Morse, adopted the singular "computer science." Morse had founded MIT's Computation Center in 1956 with a donated IBM 704 and had persuaded NSF to support the facility as a shared computer center for forty colleges in the region.[41] He chaired the organizing committee for a 1960 conference of computer center directors, also funded by NSF, and proclaimed in his report on the conference that "Computer science is a new scientific field" and "Computer science is a discipline in its own right."

38. L. Fein. 1959. The role of the university in computers, data processing, and related fields. *Commun. ACM*, 2(9): 11. For more on Fein see Joseph November's chapter 3 in this volume.

39. Ibid.

40. S. Gorn. 1963. The computer and information sciences: A new basic discipline. *SIAM Rev.*, 5(2): 150.

41. D. Walden and T. Van Vleck, editors. 2011. *The Compatible Time Sharing System (1961–1973): Fiftieth Anniversary Commemorative Overview*. IEEE Computer Society, pp. 1, 6 and 4.21; F. J. Corbató. 1989. Interview by Arthur Norberg, Charles Babbage Institute, OH 162, p. 5. Available at http://purl.umn.edu/107230.

Morse invoked intellectual sophistication as the hallmark of a scientific discipline: "The problems associated with exploiting fully the potentialities of present and projected computers are difficult and intellectually challenging."[42] George Forsythe, who spearheaded the creation of a Division of Computer Science within the Stanford mathematics department in 1961, also favored the singular term, giving a lecture that year that described "a coherent body of technique, which I call computer science."[43]

ACM leaders also attempted to change the name of their society to sound more scientific. ACM stands for "Association for Computing Machinery," a phrase that irked many theoretical computer scientists because it invoked the computer as artifact rather than research field. In 1964, Perlis' Commission of Thoughtful Persons recommended, "A new name should be found which does not contain direct reference to equipment."[44] After some discussion, in 1965 ACM held a vote of its members on a proposal to change the society's name to "Association for Computing and Information Sciences."[45] A substantial 63% of the votes favored the name change, but this fell just short of the two-thirds required, and the issue was never again brought to a formal vote.[46]

Perhaps the most explicit defense of computer science directed toward the skepticism of mainstream scientists was a 1967 letter to the editor of *Science* by three leading researchers: Allen Newell, Alan Perlis, and Herbert Simon. These three had successfully established a department of computer science at the Carnegie Institute of Technology in 1965, but evidently they still did not feel accepted by their scientific peers. Their letter begins bluntly, "Professors of computer science are often asked: 'Is there such a thing as computer science, and if there is, what is it?"' The first answer they offer is cheekily brief: "Wherever there are phenomena, there can be a science to describe and explain those phenomena. . . . Ergo, computer science is the study of computers."[47] But sensing that this "simple answer"

42. P. M. Morse. 1960. Report on a conference of university computing center directors (June 2–4 1960). *Commun. ACM*, 3(10): 520–521.

43. Forsythe's lecture, "Educational implications of the computer revolution," is quoted in D. E. Knuth. 1972. George Forsythe and the development of computer science. *Commun. ACM*, 15(8): 722.

44. A. J. Perlis. 1964. Report of the commission of thoughtful persons to the ACM council, 24 April 1964. *Commun. ACM*, 7(8): 508.

45. G. E. Forsythe. 1965. Official ACM. *Commun. ACM*, 8(7): 424.

46. G. E. Forsythe. 1965. "President's letter to the ACM membership," *Commun. ACM*, 8(9).

47. A. Newell, A. Perlis, and H. A. Simon. 1967. Computer science. *Science*, 157(3795): 1373–1374.

might not sway skeptics, they go on to argue for the uniqueness and complexity of their field of study. Addressing the hypothetical objection that the computer is only a tool—"Computers, like thermometers, are instruments, not phenomena"—the authors respond, "The computer is such a novel and complex instrument that its behavior is subsumed under no other science. . . . Hence, the computer is not just an instrument but a phenomenon as well, requiring description and explanation." They conclude by emphasizing that computer scientists share a scientific ethos characterized by "passion" for research and "confidence that intelligent, persistent curiosity will yield interesting and perhaps useful knowledge."[48] As these examples show, advocates of computer science ranged widely in their notions of the essence of science and selectively emphasized whichever features best fit their own practice.

In addition to arguing directly for their field's scientific worthiness, ACM leaders pursued a discursive strategy of generating and disseminating information to back up their claims. Quantitative data on topics such as "scientific manpower" provided powerful ammunition for science policy struggles in 1960s America.[49] By creating facts (statements about reality that were accepted as true and objective) on such topics, computer scientists could shape the context in which federal policy makers would consider their requests for funding. ACM President Oettinger was forthright about the political function of quantitative studies, arguing in 1966, "Our [funding] case and that of the universities would be much stronger if reliable data about the present state of the computer industry and its future material and manpower needs were available."[50] He proposed that ACM conduct a survey to create such data, and ACM's Education Committee responded by preparing a report on the needs of education in computer science. ACM also helped fund a survey of industry demand for computer personnel.[51] These data helped articulate a national need for computer science funding by making visible a numerical gap between the projected supply and demand.

The most visible and influential data-based interventions on behalf of computer science were two publications known as the Rosser and Pierce Reports. The Rosser Report was begun by the NRC in 1962. It was nominally requested by NSF Director Alan T. Waterman, but the instigator was Philip Morse of MIT, who was seeking increased funding from NSF for a bigger computer to support MIT's work on time

48. Ibid., 1374.

49. J. Lucena. 2005. *Defending the Nation: U.S. Policymaking to Create Scientists and Engineers from Sputnik to the 'War against Terrorism.'* University Press of America, Lanham, MD.

50. A. G. Oettinger. 1966. On ACM's responsibility. *Commun. ACM*, 9(4): 246.

51. A. G. Oettinger. 1966. President's letter to the ACM membership. *Commun. ACM*, 9(12): 839.

sharing. Arthur Grad, who oversaw NSF's computer facilities grants, advised Morse that his request was too large and that he would need to make a policy case for it.[52] Morse had been elected to the NAS in 1955, and he used his influence there to request an NRC report on computing needs—demonstrating once again how institutional access could enable discursive boundary-work. The resulting report, published in 1966, was titled *Digital Computer Needs in Universities and Colleges,* but it became known as the Rosser Report after lead author J. Barkley Rosser, a theoretical computer scientist at the University of Wisconsin.[53]

The Rosser Report proclaimed, "Computer science is rapidly assuming the position of an established discipline" and argued that computer centers were needed both to provide services for other sciences and to support computer science itself.[54] Numerous statistics were offered to justify these arguments: the number of university computing centers had increased tenfold from 1957 to 1964; "campus expenditures for computing were doubling every two years"; "35,000 computer staff positions were being created each year in the United States."[55] The report recommended doubling federal support for campus computing, to support both computer science research and training for industry computer jobs.[56] Reaction to the report was mixed, and it did not immediately result in a new funding vehicle

52. Grad came to the mathematical sciences section of NSF in 1959 and administered the computer facilities grants. He recalled the genesis of the Rosser Report as follows: "That all started with Phil Morse at MIT. They needed a bigger computer. They estimated they would need about ten million dollars. And I told them, well, there wasn't much I could do about it since my entire budget was only five. And I suggested to him that probably the best thing he could do was to have a National Academy study done pointing out the need for more money for computers. So the Academy duly appointed the committee to make those studies. . . . But it all started from Phil Morse's need for a big computer" (A. Grad. 1990. Interview by William Aspray, Charles Babbage Institute, OH 216, p. 10. Available at http://purl.umn.edu/107338). Thomas A. Keenan recalled that "the National Academy of Sciences decided to (mainly at P. Morse's instigation) to set up a committee to study the uses of computers in colleges and universities" and that Morse had asked Rosser and Keenan to serve on the committee. Keenan later came to NSF as a program officer for OCA in 1969.

53. J. B. Rosser. Chair, National Research Council Committee on Uses of Computers. 1966. Digital computer needs in universities and colleges: A report. National Academy of Sciences/National Research Council, Washington, DC.

54. Ibid., 123, 16.

55. Quoted in W. Aspray and B. O. Williams. 1994. Arming American scientists: NSF and the provision of scientific computing facilities for universities, 1950–1973. *IEEE Ann. Hist. Comput.*, 16(4): 64.

56. Rosser, Digital computer needs in universities and colleges, 1–2.

for computer science research, but it did make computer science more visible to policy makers and paved the way for a more influential study.[57]

The Rosser Report may have been weakened by a perception that it was solely concerned with serving the computing community, rather than the broader public interest.[58] But contemporary political culture offered computer science advocates an alternative discursive framing. Lyndon Johnson's 1964 State of the Union speech had urged the creation of a "Great Society" that would eliminate poverty, racial injustice, and urban problems, and the first item on Johnson's national agenda was "a program in education to ensure every American child the fullest development of his mind and skills."[59] Among those who heeded Johnson's call was John R. Pierce, an electrical engineer and research director at Bell Telephone Laboratories who was serving on the President's Science Advisory Committee. In 1966 Pierce recruited a panel of experts from universities and industry research labs to investigate national needs and opportunities in educational computing. The committee's findings were published in February 1967 as *Computers in Higher Education*, commonly known as the Pierce Report.

The Pierce Report embedded computer researchers' aspirations to be accepted as scientists in a new narrative that positioned funding for computer science as a path to national wellbeing. Its first pages boldly declared, "Adequate support of computing as a part of education is essential for a rapid and full realization of the social and economic benefits of computing."[60] The report recommended that all universities should provide educational computing and that the federal government should share the cost. This plea for funding was woven into a tale of enlightened progress that framed access to computers as almost a civil right:

> Happily, at some fortunate and forward looking colleges and universities the educational use of computers is widespread and effective. But this does not apply to

57. Aspray and Williams discount the influence of the Rosser Report (Arming American scientists, 66). Gupta, however, writes that the Rosser Report "had considerable impact on funding for computers in higher education" (G. K. Gupta. 2007. Computer science curriculum developments in the 1960s. *IEEE Ann. Hist. Comput.*, 29(2): 52).

58. Rosser's main co-author on the report, Thomas A. Keenan of the University of Rochester, felt the report's focus on funding may have been counter-productive: "I think the Bureau of the Budget at the time thought it sounded . . . self-serving" (T. A. Keenan. 1990. Interview by W. Aspray, Charles Babbage Institute, OH 217, p. 4. Available at http://purl.umn.edu/107401).

59. L. B. Johnson. 1964. State of the Union addresses by Lyndon B. Johnson. In J. Manis, editor. *An Electronic Classics Series Publication*, p. 17. Pennsylvania State University, Hazleton, PA.

60. J. R. Pierce. 1967. *Computers in Higher Education: Report of the President's Science Advisory Committee*, p. 4. US GPO, Washington, DC.

> the majority, where computing facilities are often absent or inadequate. . . . Can this deficit be remedied, so that no American need have second-rate education in this respect?[61]

Having made a case for funding *educational* computing, the report deployed this as a rationale for supporting computer *science*. After all, trained faculty would be needed to teach computing to all of these new students:

> The whole success of educational computing . . . depends on expanded education and research in computer sciences. This education requires a good faculty and access to very good computing facilities for both course work and research. We recommend that the Federal Government expand its support of both research and education in computer sciences.[62]

This recommendation, with its repeated emphasis on "research" and "science," went well beyond the basic teacher training that might have sufficed to support computer literacy courses in schools. The report also broadened the importance of academic computer science beyond its contribution to education—the ostensible focus of the study—by arguing, "In order to provide the computer experts who . . . are so vital to our national defense, to increasing our productivity, and to improving our standard of living, we must have excellent computer science departments at a number of schools."[63] At the same time, the report tried to distance computer science from its traditional, low-status service role, stating, "The computer sciences faculty should not be burdened with the administration of a computer center."[64] By invoking "research" and intellectual "excellence," the report discursively placed computer experts among the company of autonomous scientists, rather than mere technical or educational assistants. Computers were a vital policy tool, but could only fulfill that promise as the center of a new scientific discipline.

2.5 Success: The Creation of NSF's Office of Computing Activities

The Pierce Report's persuasive rhetoric—and its authors' access to the President—brought results.[65] Immediately after its publication, President Johnson told the nation in a February 1968 speech that "One educational resource holds exciting

61. Ibid., 2.

62. Ibid., 5.

63. Ibid., 22.

64. Ibid., 41.

65. Milt Rose, who was responsible for the computing programs in the NSF Mathematical Sciences Division in the 1960s, argued that it was the Pierce Report and Pierce's contacts with the

promise for America's classrooms: the electronic computer."[66] He then directed the National Science Foundation to work with the Office of Education to establish a new program for computers in education. How to implement this directive was not clear, since neither agency had a plan for educational computing, but the initiative quickly defaulted to NSF as the agency with more computer expertise.[67] NSF Director Leland Haworth established an Office of Computing Activities in July 1967 and appointed Milton E. Rose, who had been leading NSF's Mathematical Sciences section, as its first head. Commenting on the newly created OCA, Johnson predicted with satisfaction, "The day is not far when these exciting new machines will be contributing to the education of our people."[68] But OCA was not solely focused on applying computers to education: following the Pierce Report's logic to its conclusion, OCA was also given a mandate for "providing Federal leadership" in the research side of computing.[69] This fact went unremarked by the politicians, but would prove to be a turning point for computer science at NSF.

Computer science's newly raised status came with additional funding: OCA's initial budget was $22 million, a 73% increase from the $12.7 million allocated for computer activities in education and research in the previous year.[70] Perhaps as

---

president's science adviser and other high-level government officials that provided the political influence to open a computer office and embark on a major computer education initiative in the foundation (W. Aspray and B. O. Williams. 1994. Arming American scientists: NSF and the provision of scientific computing facilities for universities, 1950–1973. *IEEE Ann. Hist. Comput.*, 16(4): 68, note 15).

66. L. B. Johnson. 1967. Special Message to Congress: "Education and Health in America," February 28, 1967, in *Lyndon B. Johnson: 1967: Containing the Public Messages, Speeches, and Statements of the President*, p. 251. Office of the Federal Register, Washington, DC.

67. NSF staff involved in the startup of OCA recall that the motivation for placing the educational computing activity within NSF, rather than Education, was partly a matter of expertise, but also economic. Rose recalled in an interview, "The Department of Education was considered as the proper agency for this work on computers in education, but the Bureau of the Budget and presidential adviser Joseph Califano believed that the DOE would not have the technical command of the subject to use prudently and effectively the substantial funding" (quoted in Aspray and Williams, Arming American scientists, 62). Kent Curtis, who joined NSF in the summer of 1967, believed that NSF was given responsibility for the education program because the government was under financial pressure due to the Vietnam war, and having NSF repurpose its existing computer science program was seen as the cheapest option (K. Curtis. 1987. Interview by J. Minker, pp. 22–23).

68. L. B. Johnson. 1968. Message to the Congress transmitting annual report of the National Science Foundation, March 20. In *Lyndon B. Johnson: 1968–1969: Containing the Public Messages, Speeches, and Statements of the President*. Office of the Federal Register, Washington, DC.

significant was the new location of computer science within the overall NSF organization: rather than being under one of the research directorates, OCA reported directly to the NSF Director. This positioning may have reflected the lingering view that computer science was primarily a form of scientific infrastructure, rather than a discipline; but it also fulfilled the hopes of ACM activists by bringing computer science out from under the shadow of mathematics, where its status as a research field had always been in question.[71] It also kept computer science out of the Engineering Division, whose Advisory Committee had been lobbying since 1965 to be put in charge of computing activities.[72] Thus, OCA's organizational location gave the nascent computer science field some breathing room to develop independently of its two parent disciplines.

OCA's internal organization reflected the multiple, still-evolving identity of computer science within NSF. It had three sections that corresponded to its three functions of supporting computers as scientific infrastructure; computers as a tool for education; and computer science as a discipline.[73] These multiple identities sometimes caused tension, reflecting the uneasy position computer science still occupied between handmaiden and autonomous science; but once an independent office for computing had been established, its managers could, and did, look for opportunities to shift the balance of activities toward research.

69. National Science Foundation. 1984. Organizational development of the National Science Foundation (NSF Handbook Number 1), p. 71. NSF, Washington, DC.

70. National Science Foundation. 1968. *Annual Report of the National Science Foundation, FY1968*, Table 2, pp. 8–9. NSF, Washington, DC.

71. Weingarten recalled that OCA was put under the Director "because the program was infrastructural" (F. Weingarten. 1990. Interview by William Aspray, Charles Babbage Institute, OH 212, p. 18. Available at http://purl.umn.edu/107706).

72. Belanger notes, "In 1965 the NSF Advisory Committee for Engineering had strongly recommended that the Engineering Division 'have an important part, or even the major role, in the broad area of computers'." She implies that the placement of the Office of Computing Activities under the NSF Director, and its emphasis on education rather than engineering, was a disappointment for NSF's engineers (D. O. Belanger. 1998. *Enabling American Innovation: Engineering and the National Science Foundation*, p. 171. Purdue University Press, West Lafayette, IN.)

73. Most of OCA's initial activities and managers were simply transferred from the Mathematical Sciences section. The Institutional Computing Services section took over the role of funding universities to purchase computers as a tool for scientists. Rose recruited Kent Curtis from Berkeley to run this section, and Glenn Ingram moved over from Mathematical Sciences to serve as Curtis's deputy. Arthur Melmed was brought from Mathematical Sciences to head the Special Projects Section, also known as Computer Innovations in Education, a new program created to fulfill OCA's educational mandate. OCA's third section, Computer Science Education, Research &

Having its own office at NSF also gave computer science greater visibility and legitimacy in the larger scientific community. For example, the National Academy of Science's 1968 report on *The Mathematical Sciences* interpreted the existence of "a separate Office of Computing Activities" at NSF as evidence that the government now viewed computer science as "an independent entity."[74] Computer science also became more visible in NSF's official publications. NSF's annual reports had barely mentioned computer science until 1968, when the establishment of OCA was discussed over several pages.[75] The 1969 report went further and proclaimed "the emergence of computer science as an academic discipline."[76] Once again, institutional and discursive boundary-work went hand in hand.

2.6 Conclusion

Through a variety of rhetorical and organizational strategies, computer science advocates were able, within the space of a decade, to claim scientific respectability for their field and gain control of their own funding stream within NSF. This was a remarkable coup for such a young field. ACM leaders advanced this cause in multiple ways: by working to get representation for computer scientists on scientific advisory bodies; by generating data to create supporting facts and narratives for their cause; and by making the case to their fellow scientists for the scientific legitimacy of their field.

The story of ACM and NSF also provides a window on the meanings of *science* and *discipline* in 1960s America. Because computer scientists were forced to articulate and defend their scientific identity, we are able to witness the variety—and glaring lack of consensus—among contemporary definitions of what makes a science. Computer professionals could argue that they were scientists by virtue of having a unique subject domain within the natural world, a high level of intellectual challenge, a body of theory or technique, or even a passion for research. We also see

---

Training, which focused on computer science as its own discipline, was led by Fred Weingarten. See National Science Foundation. 1984. Organizational development of the National Science Foundation (NSF Handbook Number 1), p. 71. NSF, Washington, DC.

74. Committee on Support of Research in the Mathematical Sciences (COSRIMS). 1968. "The mathematical sciences: A report," p. 95.

75. National Science Foundation. 1968. *Annual Report of the National Science Foundation, FY 1968*, p. 196. Only two prior annual reports had mentioned computer science as a research area: the 1964 report had an extended discussion of NSF-funded time sharing research (pp. 25–26), and the 1963 report made a passing reference to computer science research in "artificial intelligence, pattern recognition, etc." (p. 12).

76. Ibid., FY 1969, p. 94.

that the notion of *discipline* can have multiple functions. In the pursuit of power, the scientific discipline as a political actor (represented in this case by ACM) can perform boundary-work to meet external challenges. In the pursuit of meaning, assembling a disciplinary identity can build a sense of value and shared purpose among practitioners of a new field.

For computer scientists in the early 1960s, recognition as a discipline could be seen as a criterion for membership in a powerful body, a measure of intellectual respectability, or a sign of internal agreement on goals and methods. The creation of the NSF Office of Computing Activities represented their success in achieving the first two forms of discipline by the late 1960s. Whether they would achieve the third remained an open question.

2.7 Acknowledgments

This work was supported through a 2012 Fellowship in ACM History. I am indebted to Bill Aspray for sharing research materials on the history of NSF's computing efforts and to the Charles Babbage Institute for access to its archival and oral history collections.

3

George Forsythe and the Creation of Computer Science As We Know It

Joseph November

3.1 The Man Who Would Remove the "M" from the ACM

If George E. Forsythe (1917–1972) had only managed to persuade a couple hundred more ACM members to see things his way, you would be reading this chapter in a book published by the Association for Computing and Information Sciences instead of the Association for Computing *Machinery*. In 1965, while serving as president of the ACM, Forsythe had called on the organization to change its name to reflect the importance of "the art and science of representing and processing information" to the society. Ultimately, 63% of the roughly 6,000 ACM members who voted on the issue favored changing the organization's name, but the measure did not quite meet the two-thirds majority necessary to pass. "Personally, I am very sorry," wrote Forsythe, "This is the first proposed change that the membership has ever voted down."[1]

In the decades since 1965, the ACM continues to retain its original name, but Forsythe's work to reorient the organization towards promoting a "science of computing" left a powerful legacy, both for the ACM and for the broader field of computer science. Examining how Forsythe arrived at his vision for computer science and how his vision became so influential can give us much insight into why the field of computer science took on its distinct form—and the ACM's role in that formation process.

1. G. E. Forsythe. 1965. President's letter. *Commun. ACM*, 8(9): 541.

If you are not a computer scientist, you probably have not heard of George Forsythe; but within that field, especially the parts of it connected to the influential Stanford Computer Science Department, which he founded in 1965, he is a towering historical figure. Stanford computer scientist Donald Knuth, the creator of TeX and the author of *The Art of Computer Programming*, summarized Forsythe's status: "It is generally agreed that he, more than any other man, is responsible for the rapid development of computer science in the world's colleges and universities."[2] Another Stanford computer scientist, John Herriot, shared such sentiment, arguing, "that the fact that nationally as well as at this university, computer science came to be recognized as an area of study that was of importance in itself, was due to the efforts and the vision of George Forsythe more than any other person. . . . "[3]

Indeed, Forsythe's name adorns some of the most crisply articulate and widely cited early writings about the nature and aims of academic computer science, many of which were published in *Communications of the ACM* when (and immediately after) he served as president of the organization from 1964–1966. And in his honor, the first home of Stanford University's Computer Science Department, Forsythe Hall, was named for him. The building still bears his name even decades after the department he founded outgrew it.

More than 40 years after Forsythe died, tales of his exemplary brilliance, generosity, and vision abound. His place in academic genealogies and bibliographies is, if anything, becoming more prominent as those whom he trained (and those with whom he worked) achieve intellectual and institutional success.[4] Decades on, strong feelings remain about Forsythe's efforts to establish computer science as an academic discipline, and he is often cast as a martyr-like figure whose vision of what a computer science department ought to do was at once widely emulated but also left unfulfilled when Forsythe died before he could fully implement it. Although many of the accounts of Forsythe's life show him as industrious and academically productive, few delve deeply enough into either his published or private writings to

---

2. D. E. Knuth. 1972. George Forsythe and the development of computer science. *Commun. ACM*, 15(8): 721–727.

3. Oral history interview with J. Herriot, Charles Babbage Institute, OH 21. Available at http://purl.umn.edu/107356.

4. See N. H. F. Beebe. 2012. A Bibliography of publications of George Elmer Forsythe. Available at ftp://131.254.254.45/pub/netlib/bibnet/authors/f/forsythe-george-elmer.pdf. Cleve Moler's 1997 "Tree of Forsythe Students," shows the institutional placements of Forsythe's graduate students. Available at http://infolab.stanford.edu/pub/voy/museum/forsythetree.html.

learn about the world out of which computer science emerged and how Forsythe's often frustrated efforts to overcome the challenges presented by that world grew into many of the specific characteristics of computer science today.

Initially, my research centered on the questions of how, specifically, Forsythe built and promoted his vision for computer science and why that vision was apparently so well received. Ironically, the vast amount of available archival and other less traditional documentary material have muddied the issue to the point that a neat synthesis is impossible in the space of a book chapter. My hope had been that, by drawing from the collected documents related to Forsythe's work, I would be able to lift the shroud of mythology surrounding the man, a mythology that has grown out of narratives centered on his unexpected death in 1972 at the age of 55 and at the height of his career. However, in the process of trying to moving beyond pronunciations like "one might almost regard him as the Martin Luther of the Computer Reformation!"[5] I discovered there was another shroud, one formed from the presence of *too much* documentary material.

As cultural anthropologist Diana E. Forsythe (1947–1997), the daughter of George and Alexandra (Sandra) Forsythe, explained, her father "documented his every professional act."[6] Much of this writing has found its way into the Stanford University Archives, and particularly (but not exclusively) into the George and Alexandra Forsythe Papers (1936–1979), which includes thousands of letters, memos, notes, drafts, and diaries (as well as formally published material) and which spans 40 linear feet. She added that the experience of going through the collection was "painful," and that the "letters and memos in his files confront me again and again with reminders of the plans and dreams curtailed by his sudden death."[7]

Beyond showing what George Forsythe hoped computer science could be, his notes include candid reflections on the many institutional, personal, and intellectual challenges faced by early promoters of computer science. Given the volume and complexity of Forsythe's writing, a book-length scholarly biography would be necessary to enable readers to most productively make use of his perspective on the

5. Knuth, George Forsythe and the development of computer science, 721–727.

6. D. E. Forsythe. 2001. George and Sandra's daughter, Robert's girl: Doing ethnographic research in my parents' field. In D. J. Hess, editor. *Studying Those Who Study Us: An Anthropologist in the World of Artificial Intelligence*, pp. 183–196, on 191. Stanford University Press, Stanford.

7. Ibid., 191.

formation of computer science.[8] The good news is that by glancing at selections of the rich material written by—or about—Forsythe, we can nevertheless gain access to some otherwise unreachable foundations of computer science.

3.2 Forsythe Before CS: From Mathematics to Meteorology to Computing

When Forsythe began his career at Stanford in 1957 there were very few people using general-purpose electronic digital computers, and there was no such thing as a computer science department.[9] Forsythe was at the time a formally trained mathematician (by Jacob Tamarkin at Brown University) specializing in numerical analysis who had a strong interest in electronic computers. Although he had initially been on track to pursue a career in mathematics, his course changed during World War II when he joined the US Army Air Force (USAAF), where he worked to develop mathematical solutions to meteorological problems. There he encountered the work of British mathematician and meteorologist Lewis Fry Richardson, who had predicted that someday machines capable of computing differential equations could be used to forecast weather. Besides seeing computing machines as being useful for predicting the weather, Forsythe began to believe by the late 1940s that computers could broadly transform science and industry.[10]

During the 1950s, Forsythe worked as an applied mathematician for the Institute for Numerical Analysis at the National Bureau of Standards. The Institute's computer, the Standards Western Automatic Computer (SWAC), interested him and his mathematician wife, Sandra, enough that they followed its construction closely and learned how to program it. By the mid-1950s, the Forsythes were publishing algorithms they had developed to get SWAC to efficiently work with algebraic equations. This work captured the attention of Stanford mathematician John G. Herriot, a longtime friend and fellow Tamarkin student, who recruited him to Stanford's Mathematics Department to help train graduate students to use com-

8. Speaking about the Forsythe papers, "Knuth suggested that Forsythe's notes could form the core of an intellectual history of computer science through 1972." See P. Davis. 1998. Remembering George Forsythe. *SIAM News*. Available at http://www.siam.org/news/news.php?id=805.

9. Technically, Forsythe began his *second* career at Stanford in 1957; he had briefly worked there as a mathematics instructor in 1941, before he joined the USAAF as a meteorology researcher and instructor.

10. J. November. 2007. George Elmer Forsythe. In *The New Dictionary of Scientific Biography*. Charles Scribner's Sons, New York.

puters in their research.[11] By 1960, Forsythe had co-authored a textbook, titled *Finite-Difference Methods for Partial Differential Equations*, which contained much discussion of computer use, but it cast computers as *tools* to be used in service of mathematics rather than objects worthy of study in their own right.[12]

In 1961, as Forsythe sought to create ever-more efficient algorithms for carrying out numerical analysis work on computers, he came to the conclusion that specialized training in computer operations would be necessary to provide "the general-purpose mental tools" mathematicians needed to break new theoretical ground. With Herriot's help, he founded the Division of Computer Science within the Mathematics Department. The curriculum he established there included not only traditional mathematical training in numerical analysis and automata theory, but also courses in programming, data processing, game theory, information theory, information retrieval, recursive function theory, and computer linguistics.[13]

3.3 Hard Lessons on the Road to Computer Science

Stanford University's Computer Science Department has been remarkably productive. Its faculty and graduates have consistently been leaders of the field, and it is widely celebrated as a model of how a good department ought to work. However, the department's continued existence as a separate department (first in the School of Humanities and Sciences then in the School of Engineering)—five decades after Forsythe established it in 1965—is a sign that some of his most ambitious plans have gone unrealized. To understand how such a successful department's longevity contradicts Forsythe's long-term hopes for computer science, one must become familiar with the environment in which Forsythe was working in the 1960s and with his intellectual agenda as well as the challenges he faced. In short, the entity known as a "computer science department" was but a step—albeit a necessary one—toward Forsythe's plans to make working with and studying electronic digital computers truly productive. Seeing why that is the case requires a trip back to when the notion of investing heavily in computing was not popular at Stanford.

11. Ibid.

12. G. Forsythe and W. Wasow. 1960. *Finite-Difference Methods for Partial Differential Equations*. Wiley, New York.

13. G. E. Forsythe. June 25, 1965. Stanford University's program in computer science. Technical Report CS26, Computer Science Department, Stanford University. Available at http://i.stanford.edu/pub/cstr/reports/cs/tr/65/26/CS-TR-65-26.pdf).

Forsythe's first effort, carried out in 1962, to generate broad institutional support—and raise enough funds to acquire an IBM 7090—for the Division of Computer Science ended in failure, and its lessons would shape his many successes in the years to come. What Forsythe had hoped to do was to expand Stanford's Computation Center, which was run by the Division of Computer Science, by promoting it as an accessible place where scientists could go to find ways to employ computers in their research. Forsythe believed that the scientists who would benefit the most from using computers were biologists, and he canvassed the Stanford campus in an effort to convince them to use computers. He had been "firmly convinced that computing in life science research will ultimately be an absolutely essential ingredient for the rapid growth of fruitful investigations."[14] Further, even though his areas of expertise, numerical analysis and meteorology, were far removed from the concerns of biologists, he believed he could create a productive rapport with biologists because his father, Warren Ellsworth Forsythe, had been a prominent medical research professor at the University of Michigan.[15]

At first the effort went well, with Forsythe garnering commitments from diverse biology and medical researchers, including pharmacologist Keith Killiam, neurologists Kao Liang Chow and Frank Morrell, psychologist Karl Pribram, and biostatistician Lincoln Moses.[16] In the coming decades, all of these scientists would become known as pioneering computer users in their respective fields. In the early 1960s, though, their enthusiasm alone was not enough to impress the entities from which Forsythe sought support, namely his own department, Stanford's administration, and the National Institutes of Health (NIH).

On Stanford's campus, Forsythe's effort to gain funding for the Computation Center could not overcome the skepticism about computing that had arisen in reaction to a 1956 study conducted by Louis Fein, a non-academic computing consultant, on the prospects of building a computer science department or school at the university. In short, Fein recommended establishing "computer-related science" as a "supra-discipline," one that he called "Synnoetics," under the aegis of which would be united "Information Theory . . . Automation . . . Cybernetics . . . Operations Research . . . Theory of Games . . . Data Processing . . . Artificial Intel-

14. G. E. Forsythe. December 4, 1962. Memorandum. "NIH." Stanford University, ACME Collection, SC 236 Box 1 Folder 2.

15. "Warren Ellsworth Forsythe," Faculty History Project, University of Michigan. Available at http://www.lib.umich.edu/faculty-history/faculty/warren-ellsworth-forsythe.

16. J. November. 2012. *Biomedical Computing*: *Digitizing Life in the United States*, p. 230. Johns Hopkins, Baltimore.

ligence . . . Management Science . . . Bionics. . . . "[17] Fein recalled that his report was viewed by faculty and administrators as too ambitious. John Herriot, for instance, according to Fein, violently dismissed it: "[Herriot] would get up and shout, 'What you want is pie in the sky and you can't have pie in the sky!' And that was the reception I got."[18] Fein also recalled that even the newly hired George Forsythe's attitude towards the report was "equivocal; he could go for it or not go for it."[19] From Stanford Provost Fred Terman's perspective, "the faculty were always coming to him with one hot project after another and this was just another hot project as far as he was concerned. He couldn't see that this was any better than any other hot projects circulating in the air."[20] Although Terman is so often cast as a great promoter of computer science, there was a time back in the early 1960s when it was clear that he had yet to be convinced of its exceptional promise.

Within the mathematics department, Forsythe's colleagues, including Herriot (who had personally recruited him), distanced themselves from efforts to expand the Computation Center, on the grounds that the activity was at once too ambitious and too pedestrian. The department's chair, David Gilbarg, as Fein remembered in 1979, "thought computing was like plumbing. We don't have plumbers studying within the university and what does computing have to do with intellect? I am exaggerating of course. He was murder on it and he may still be to this day."[21] Forsythe agonized over this unexpected criticism and his colleagues' apparent fears about how getting involved in computing could lead to losses of status and autonomy. Renowned technology writer Pamela McCorduck, who in the 1970s examined Forsythe's private writing on the matter, succinctly described the situation: "You really see that laid out in Forsythe's papers—the tearing of hair over what the relationships between the department and the center should be—I mean it is just a non-issue nowadays but then it seemed to be very difficult."[22] These kinds of arguments against academic computing were not lost on Forsythe and he later addressed them publically and robustly.

17. Fein reproduced much of his 1956 report in L. Fein. June 1961. The computer-related sciences (Synnoetics) at a university in the year 1975. *Am. Sci.*, 49(2): 149–168.

18. Oral history interview with Louis Fein. Charles Babbage Institute, OH 15, p. 11. Available at http://purl.umn.edu/107284.

19. Ibid.

20. Oral history interview with Albert H. Bowker. Charles Babbage Institute, OH 6. Available at http://purl.umn.edu/107140.

21. Oral history interview with Louis Fein, 11.

22. Pamela McCorduck in Louis Fein oral history interview, 12.

More damaging in the short term to Forsythe than being lumped together with Fein was the negative reaction from the federal agency from which he had sought external funding, the NIH. In his 1962 grant proposal to the NIH, Forsythe claimed that after the facility had been tested by the first wave of researchers it would reach enough new users that it could establish "the foundation for a future general medical data processing center embracing all aspects of medical problems." Forsythe further argued that the facility would "establish an intellectual bridge between the computation facility and the medical center, bringing into focus the problems of quantification peculiar to biological systems and also serve to build a reservoir of talent who would be familiar with modern computational facilities on the one hand and the kinds of problems facing medicine on the other." Toward these ends he had requested $1 million to be spent over five years.[23]

Over the last six months of 1962, NIH reviewers sent Forsythe mixed signals. Some had been swayed, but most were "unwilling to game so much money in equipment and personnel on an 'unproved' approach to the general problem of processing medical data." One sticking point was Forsythe's plan for the Center to carry out mass conversions of analog data into digital data. Up to that point, no such system ever had been successfully implemented in a biomedical environment. Further, in other contexts similar (and more costly) efforts had proved unproductive, such as the large-volume analog-to-digital conversion projects undertaken by McDonnell Aircraft Corporation and the (NIH-funded) UCLA biomedical computation center.[24] Sensing that a rejection was looming, Forsythe scrambled to submit an amendment to his proposal, which he called a "counter-proposal," and which included provisions to train Stanford life scientists in enough mathematics to prepare their analog data for conversion to digital. He also offered to use some of his funding to give the Stanford scientists access to existing analog-to-digital conversion facilities, such as those McDonnell and UCLA.[25]

In November 1962, the NIH formally rejected all forms of Forsythe's proposal. For Forsythe, the "complete refusal of an extremely well thought-of plan," was dispiriting, especially due to the tremendous effort he and others in the Division of Computer Science had put into the project: "I don't believe any proposal or report

23. G. E. Forsythe and K. Killam. Medical Data Processing, Stanford University Archive, ACME Collection (SC 236), Box 1, Folder 2, p. 5.

24. Ibid.

25. G. E. Forsythe. November 29, 1962. Memorandum. "NIH Grant." Stanford University, ACME Collection, SC 236, Box 1 Folder 2. In this grant, Forsythe made no distinction between corporate and academic computer resources.

we have ever made was so much thought about, or involved coordination with so many people."[26] Initially, Forsythe had been "bitter about the matter," but as he pushed the NIH review committee for details, though, the agency's justification for turning down his grant proved revelatory.[27]

One basis for rejecting the grant, NIH reviewers noted, was that for all the intellectual merit of Forsythe's vision for biomedical researchers to use computers, there were few means to publish much of the work that would be done. Mathematics and engineering journals, let alone biology and medical research journals, did not have provisions for publishing computing techniques and algorithms, regardless of how useful they might prove. This would be problematic when it came to any of the NIH's future decisions related to continuing Forsythe's funding or that of researchers who had committed to computing, because the agency tended to measure success of grants in terms of the quantity and prestige of publications they generated.[28]

The other, more surprising, criticism coming from the NIH was that Forsythe had not done enough to promote computing as a productive intellectual activity to scientists. Forsythe had gathered a diverse group of researchers to use the Computation Center, but what about the larger population that had turned him down? Forsythe's initial reaction had been that he and other Division of Computer Science faculty and staff had already been proselytizing the good news of computing: "All of our staff members, the Director and Associate Director included, act both formally and informally as computing missionaries around the campus. Much of this missionary work is accomplished through personal contact. . . . "[29] However, as Thomas J. Kennedy, Jr., one of the members of the NIH committee that had rejected Forsythe's proposal, saw it, Forsythe had made a good intellectual case for computing but had failed to provide scientists with the resources to overcome the "'cultural' problems associated with introducing computer technology into the main stream of life science research."[30]

26. Ibid.

27. R. Wade Cole correspondence to Thomas J. Kennedy, Jr. (NIH), December 5, 1962, Stanford University, ACME Collection, SC 236, Box 1 Folder 2.

28. G. E. Forsythe. April 30, 1966. Memorandum. "One More Pass at NIH." Stanford University, ACME Collection, SC 236, Box 1 Folder 2.

29. Cole correspondence to Kennedy.

30. Thomas J. Kennedy, Jr. correspondence to R. Wade Cole. December 18, 1962, Stanford University, ACME Collection, SC 236, Box 1 Folder 2.

After spending the first half of 1963 futilely attempting to find ways to get the NIH review committee to change its mind and attempting to get Stanford itself to pay for the bulk of the Computation Center, Forsythe gave up. In 1965, he turned over his correspondence and notes related to the failed grant to the next person who would try to establish a biomedical computing center at Stanford, geneticist (and Nobel Laureate) Joshua Lederberg.[31] By then, Forsythe had a much clearer vision of how to get universities to take computer science seriously, and he had acquired the means to widely share that vision.

3.4 Building a Home for Computer Science at Stanford

Following the setback with Computation Center, the Mathematics Department continued to hire mathematicians interested in working with computers. As the Division of Computer Science grew, Forsythe also began to take a more extroverted approach to establishing the importance of computing. On campus, he attempted to raise awareness of computers' potential by giving lively presentations to other departments and to university administrators. He also invested much time and energy into conversing with people one-on-one about computing, even engaging students at football games by having the Computation Center print up material used for a massive "card stunt" carried out by tens of thousands of cheering fans.[32] From Herriot's perspective, this activity was crucial in gathering enough support for Stanford to continue to invest in computing.

> Well, he gave talks. He talked to the administration and he convinced them about things that should be going on. But I suppose that more than anything, giving talks at various meetings and just talking to people, making the ideas become familiar to everybody . . . George was very good at his mission, explaining things to people and he had a lot of energy. We owe a lot to George.[33]

Forsythe's cheerleading for computing played particularly well to the audience of Stanford administrators. Although Provost Fred Terman and Dean Albert Bowker had balked at the 1956 Fein Report and although they did little to help Forsythe in

31. For this reason, the papers related to Forsythe's attempt to establish a computing center for Stanford biomedical researchers were deposited by Lederberg into the *ACME* (Advanced Computer for Medical Research) Collection at Stanford's archive rather than the Forsythe papers.

32. Oral history interview with Alexandra Forsythe. Charles Babbage Institute, OH 17, p. 16. Available at http://purl.umn.edu/107291.

33. Oral history interview with John Herriot. Charles Babbage Institute, OH 21.

his effort to secure NIH funding, they both maintained a strong interest in continuing to make Stanford into an important center of activity related to computing. By 1963, Terman's plan (crafted jointly with Stanford's former President John Sterling) to reorient the university towards interdisciplinary and federally-supported research was well underway.[34] Already, Terman had overruled faculty concerns when he reoriented Stanford's medical school away from training and towards research. After dismissing, en masse, faculty who prioritized clinical training, Terman oversaw the hiring of top-notch interdisciplinary scientists, such as biochemist Arthur Kornberg and geneticist Joshua Lederberg, whose work could be applied to medical problems.[35] Under Terman's watch, the physical sciences and engineering saw such restructuring too, with faculty encouraged to collaborate across disciplines and also to raise money for their research by applying for federal grants or by finding ways to bring their work to the local university-cultivated marketplace—fostering Terman's reputation as "Father of Silicon Valley."[36] Throughout this effort, Terman increasingly saw computerization as a means to the end of transforming Stanford into a research powerhouse. Forsythe's hiring had been a consequence of Terman's emphasis on computing, as had been the hiring of other computing experts, most of whom had enough mathematics background to be housed without much controversy in the Mathematics Department.

Terman's desire to establish Stanford as a major center for computing began to create tension in the Mathematics Department when he sought to add to the Division of Computer Science faculty who were computing experts but who had little to do with professional mathematics. The tensions came to a head when Forsythe (backed strongly by Terman) attempted to recruit John McCarthy to Stanford.[37] Although McCarthy held a Ph.D. in mathematics, and although he was a champion of using mathematics to improve what would become computer science, the MIT computing expert was interested primarily in developing programming languages and exploring areas like timesharing and artificial intelligence. Other potential

34. R. S. Lowen. 1997. *Creating the Cold War University: The Transformation of Stanford*, pp. 120–146, 157–177. University of California Press, Berkeley.

35. E. J. Vettel. 2004. The protean nature of Stanford University's biological sciences 1946–1972. *Hist. Stud. Phys. Biol. Sci.*, 35(1): 95–113, on 96.

36. C. S. Gillmor. 2014. *Fred Terman at Stanford: Building a Discipline, a University, and Silicon Valley*, p. 349. Stanford University Press, Stanford, CA.

37. Oral history interview with Alexandra Forsythe. Charles Babbage Institute, OH 17.

hires, such as Edward Feigenbaum, who was trained at Carnegie Institute of Technology (now Carnegie Mellon University) by Herbert Simon and Allen Newell in areas that emphasized economics and psychology, had even less connection to mathematics.

In order to bring in McCarthy, Feigenbaum, and other computer experts whose presence Terman saw as crucial to bolster Stanford's computing, he formally established the Computer Science Department within the School of Humanities and Sciences in January 1965 and named Forsythe as department chair.[38] Some mathematicians, like Herriot, switched over to the new department. Like other Stanford departments, Computer Science had the authority to grant Ph.D. and M.S. degrees, but unlike traditional departments it did not have much in the way of undergraduate teaching obligations—the focus was on research and the training of graduate students.[39]

Terman had taken a bold step in creating the Computer Science Department, but he was able to convince Stanford faculty, administrators, trustees, and himself of the appropriateness of this move by grounding it in the justifications Forsythe had provided to him for establishing computer science as an academic discipline. Although Stanford was not the first university to found a computer science department—Carnegie, Cornell, Illinois, Michigan, North Carolina, Notre Dame, Pennsylvania, Penn State, Purdue, Stanford, Texas, Toronto, and Wisconsin all had (or were about to have) them—Forsythe's vision of computer science was articulated in a way that made its message resonate with a wide audience in the academic world and the (often overlapping) early world of computing.[40]

3.5 Forsythe and the Challenge of Defining "Computer Science"

When Forsythe was named department chair, he had already been elected president of the ACM from 1964–1966. As president (and later as an active ex-president) he used the *Communications of the ACM* as a pulpit to broadcast his vision of

38. G. Wiederhold, Stanford Computer Science Department timeline. Available at http://www-cs.stanford.edu/about/department-timeline (accessed February 2016).

39. G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 6. Much of Forsythe's language on graduate training found its way into another influential ACM report: W. F. Atchison, et al. March 1968. Curriculum 68: Recommendations for academic programs in computer science. *Commun. ACM*, 11(3): 151–197. See also chapter 1 in this volume by Misa and chapter 5 by Dziallas and Fincher.

40. G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 4.

academic computing. (For the evolution of the ACM's definition of "computer science," particularly as articulated by the organization's presidents, see Janet Abbate's chapter 2 in this volume.) Much of what Forsythe said and wrote to Terman (and that Terman and others in Stanford's leadership found so convincing) wound up in a more accessible, polished form in *Communications* in the mid-late 1960s.

Tellingly, other widely shared visions of computer science written around the same time did not have nearly the resonance as those of Forsythe, despite sometimes appearing in publications with much larger readerships than *Communications.* Defining "computer science" in a letter published in *Science* in 1967, shortly after Forsythe published his own views in *Communications*, Allen Newell, Alan Perlis, and Herbert Simon (all renowned computing figures by then) emphasized computers themselves as objects of study. They wrote: "Phenomena breed sciences. There are computers. Ergo, computer science is the study of computers."[41] They dismissed notions that computers could be thought of as mere instruments ("like thermometers") or that the study of these devices could be "subsumed under any existing area of science." Making the case for centering computer science on the study of machinery, they assailed the notion that work with programs and algorithms should instead be the central activity of computer science. They even went so far as to issue a veiled criticism of Forsythe's popular 1965 attempt to rename the ACM as the Association for Computing and Information Sciences, writing: "Showing deeper insight than they are sometimes credited with, the founders of the chief professional organization for computer science named it the Association for Computing Machinery."[42] Forsythe's own writing, though limited to the much smaller readership of *Communications*, took a more nuanced, inclusive approach to the issue of which phenomena computer scientists might study, and unlike Newell, Perlis, and Simon's letter, discussed the pragmatic aspects of accessing those phenomena.

Besides being practically grounded and widely encompassing, Forsythe's vision eschewed much of the popular rhetoric that came up in discussions of computer science. For instance, his goals for computer science were far more modest that what *Time* magazine wrote in its 1965 cover story, "The Cybernated Generation,"

41. A. Newell, A. J. Perlis, and H. A. Simon. 1967. Computer science. *Science*, 157(3795): 1373–1374. Forsythe would later describe this definition as "perhaps the tersest answer" to the question of "what is computer science?" See G. E. Forsythe. 1968. What to do till the computer scientist comes? *Am. Math. Month.*, 75(5): 454–462.

42. Newell, et al., Computer science, 1373–1374. Perlis was president of the ACM 1962–1964 immediately before Forsythe.

about Louis Fein's vision for computer science serving as the means to free other scientists from the shackles of laborious calculating and organizational work:

> 'What the hell are we making these machines for,' says Dr. Louis Fein, a California computer consultant, 'if not to free people?' Many scientists hope that in time the computer will allow man to return to the Hellenic concept of leisure, in which the Greeks had time to cultivate their minds and improve their environment while slaves did all the labor. The slaves, in modern Hellenism, would be the computers.[43]

Fein, unlike Forsythe, also waxed eloquent about the potential of computers as means to bring more rigor to their users' thinking: "We are quite familiar with the role of computers as problem solvers, as calculators and as simulators, emulators, and imitators. But computers play their most significant role as a Socratic goad to analysis and problem formulation."[44] Such insights may elicit a knowing nod in agreement today, but in 1965 very few people, including perhaps the majority of decision-makers at universities, had had the bruising experience of attempting to transform their ideas into a functioning computer program.

These other visions may have reached much larger audiences than Forsythe's writing in *Communications*, but what Forsythe had to say about the scope and aims of computer sciences seems to have stuck. Fein's vision of "Synnoetics" never gained much traction, and for all the intellectual accomplishments of computer scientists at Carnegie (Perlis, Simon, Newell), their institutional models were not widely emulated. There was something appealing about Forsythe's approach to computer science at Stanford, an approach honed by carefully evaluating his many setbacks in earlier years. A close look at what Forsythe actually wrote to the ACM membership gives insight into that appeal.

Forsythe's best-known manifesto for computer science was published in January 1967 in *Communications of the ACM* at the conclusion of his two-year term as the organization's president. It was titled "A University's Educational Program in Com-

43. "The Cybernated Generation." 1965. *Time*, 85(14): 84-91. Available at http://content.time.com/time/magazine/article/0,9171,941042,00.html. Alluding to Greek philosophy to explain computing is a running theme of the article and of Fein's own published work.

44. L. Fein. 1963. Computer-oriented peace research. *AFIPS '63 (Fall) Proceedings of the November 12–14, 1963*, pp. 631–639, on 632. Tragically, considering his role in the origins of "computer science," Louis Fein's papers were discarded before they could be archived; see A. Leonard. June 10, 1999. Can history survive Silicon Valley? *Salon*. Available at http://www.salon.com/1999/06/10/stanford/.

puter Science," and was based on several Stanford technical reports.[45] Forsythe's article answered the question, "What is Computer Science?" in ways intended to raise awareness of the field's great potential and of the feasibility of many of the steps to reach that potential. Becoming familiar with what Forsythe wrote is essential for understanding why his vision was so appealing to so many at Stanford and beyond, and to understanding the challenges faced by computer science as the discipline emerged.

Forsythe's direct answer to "What is Computer Science?" would be uncontroversial today: "I consider computer science, in general, to be the art and science of representing and processing information and, in particular, processing information with the logical engines called automatic digital computers." He elaborated on this, again in a way that would raise few qualms in the 21st century:

> Computer science deals with such related problems as designing automatic digital computers and systems, the design and description of suitable languages for representing both processors and algorithms, the design and analysis of methods of representing information by abstract symbols, and of the complex processes for manipulating these symbols.[46]

In the mid-1960s, when Forsythe was writing the above, such answers were highly provocative, and needed to be defended with extraordinary precision and diplomacy. One issue was that many scientists, mathematicians, and engineers found the "science" component of the term "computer science" highly problematic and even offensive. Thus, Forsythe devoted a large part of his article to defining computer science as a science, albeit one that involved activities not undertaken in other scientific fields. The problem of legitimizing computer science as a science may seem quaint today, but in the 1960s amid the science spending boom, it was in the interest of academic computing specialists to have their activity viewed as *science*.

To establish computer science, and to establish how it could remain a science while distinct from other forms of science, Forsythe made a lengthy comparison of

45. G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 3–11. The article is based on G. E. Forsythe. May 18, 1966. A university's educational program in computer science. Technical Report CS39, Stanford University, Computer Science Department. That report is based, in turn, on G. E. Forsythe. June 25, 1965. Stanford University's program in computer science. Technical Report CS26, Computer Science Department. And that report derives from memos, correspondence, and talks written by Forsythe in 1963 and 1964.

46. G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 3.

it to physics. "The ultimate purpose of physics," he argued, "is the intellectual one of understanding the physical world, and the role of experiment is the secondary one of assisting in that understanding." In computer science the task was different, not in kind but rather in emphasis:

> Similarly, we may expect that someday the agreed ultimate purpose of computer science will be to understand the behavior of information and the laws which govern its processing. At the present time, however, many computer scientists consider the design of optimal processes and processors of information as the primary purpose of their subject, and relegate to theory the secondary role of suggesting techniques, methods of analysis, and proofs of optimality. Whether one is primarily interested in understanding or in design, the conclusion is that, if experimental work can win half the laurels in physics, then good experimental work in computer science must be rated very high indeed.

What Forsythe meant by "laurels in physics," were Nobel Prizes and recognition that one's activities opened up productive new lines of scientific inquiry, and he pointed to Donald Glazer's recent Nobel "solely for the design of a bubble chamber, since this opened up a new field of physics."[47] Theoretical work, he maintained, would have to wait, although only a little while: "As long as computers continue to changing drastically every three or four years, there is scarcely a chance to sit down and contemplate the creation of a theory. In this respect our subject is reminiscent of early engineering, and also of mathematical analysis in the time after Newton. I wish to emphasize my belief that this is a passing stage of computer science."[48]

If one could argue that technique in computer science was the same as it was in physics, there was also the matter of the object of study. Physics seemingly studied natural phenomena, whereas computer science studied artificial ones. Forsythe's response was that computer science's object of study was not a set of machines but rather "information—in both natural and artificial systems." This area of activity covered "information representation—as in the genetic code or in codes for efficient message transmission, and the study of information-processing devices and techniques, such as computers and their programming systems."[49]

47. G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 4.

48. G. E. Forsythe. 1968. What to do till the computer scientist comes? *Am. Math. Month.*, 75(5): 456.

49. G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 3–11. Forsythe's 1966 technical report goes into more detail about the nature of information.

Forsythe also addressed the then-pressing issue of distinguishing what he saw as good computer science, which was heavily laden with mathematics, from the kinds of work undertaken in mathematics departments. As he saw it, even though computer science could be thought of as a branch of mathematics, its priorities were sufficiently different from mathematics that the latter field had no mechanisms to reward or protect those who did meaningful work to develop algorithms and computational techniques.

In a sense, the problem was that if one regarded computer science as part of mathematics, it would, despite its extreme usefulness, be viewed by mathematicians as the lowest form of mathematics and its practitioners treated accordingly. Forsythe, who was a highly accomplished mathematician and who had received tenure on the basis of published theoretical and applied mathematical work, saw the relationship between mathematics and computer science in hierarchical terms, with theoretical—"pure" in his language—mathematics at the top and computer science at the bottom:

> Pure mathematics is interested in the *syntactical* relationship among symbols, quite apart from their meaning in the physical world or their computability.
>
> The applied mathematician is primarily concerned with the *semantics* of symbols, i.e. what do mathematical theorems mean as applied to the physical world?
>
> The computer scientist is concerned with the *pragmatics* of the applications of mathematics to problems. What algorithms can actually be used to calculate the roots of an equation? What does it cost in storage or time or human effort to perform a certain algorithm?[50]

Journals that trafficked in theorems or even their mundane applications did not provide space for researchers to publish algorithms designed to communicate those theorems to machines. Without a place to publish algorithms, there were no mechanisms for universities that hired and promoted faculty on the basis of publication to reward computer scientists embedded in mathematics departments; they would simply appear to be very unproductive mathematicians.

Besides being looked down upon by mathematicians, productive computer scientists, according to Forsythe, could provoke their ire. There would be occasions,

50. Ibid., 4, emphases in original.

he predicted, when effective computer programs could obviate the need for thinking mathematically altogther: “the computer not only permits heretofore unmanageable mathematical processes to be carried out for the first time, but it also sometimes permits the experimenter to be freed from any mathematical model whatever!”[51]

Clearly distinguishing computer science from engineering was also on Forsythe’s agenda. As was the case in mathematics, an argument could be made that computer science could be regarded as a branch of engineering. Forsythe conceded that “a central theme of computer science is analogous to a central theme of engineering science—namely, the *design* of complex systems to optimize the value of resources.” However, there was one crucial distinction between engineering and computer science: “Perhaps the main difference is that computer scientists work with a very abstract medium (information), and [they] design systems typically far more complex than most elaborate engineering systems.”[52]

If one could summarize computer science, Forsythe concluded, it would share commonalities with mathematics, engineering, and the natural sciences. In his words, “computer science is in part a young deductive science, in part a young experimental science, and in part a new field of engineering design.” This was something far beyond “merely the union of applications of a computer to diverse problems.”[53]

Considering the ways computer science was unique and considering the ways it was connected to other established disciplines, the question then arose as to where a department of computer scientists should be housed in a university. Problematically, a case could be made that computer science, at least at a conceptual level, could be connected to several disparate areas:

> Computer science must . . . concern itself with such theoretical subjects supporting its technology as coding and information theory, the logic of the finitely constructible, numerical mathematical analysis, control theory, switching theory, automata theory, mathematical linguistics, graph theory, and the psychology of problem solving. Naturally these theoretical subjects are shared by computer science with such disciplines as philosophy, mathematics, engineering, operations research, and psychology.[54]

51. Ibid., 5.
52. Ibid.
53. Ibid., 4.
54. Ibid., 3–4.

As Forsythe saw it, the solution was, first, to recognize that computer science was largely misunderstood because most observers only see part of it; and, second, to regard it as a basic liberal art, one foundational to any student's education. As a recipient of an undergraduate education at Swarthmore College, an elite liberal arts school, Forsythe knew well the benefits of a solid general education, and he hoped computer science would become a central part of that experience: "The most valuable acquisitions in a scientific or technical education are the general-purpose mental tools which remain serviceable for a lifetime. I rate natural language and mathematics as the most important of these tools, and computer science as a third.[55]

Given the above constraints and given the way universities were organized in the 1960s, Forsythe believed that computer science should be housed in "the school of letters and sciences, because of its close ties with departments of mathematics, philosophy, and psychology." He hastened to add that despite not being institutionally integrated with engineering, "the relations [of computer science] with engineering departments concerned with systems analysis and computer hardware technology should be close."[56] Stanford's Computer Science Department was established in just such a way, as were the many departments that emulated it.

What tends to be overlooked in remembrances of Forsythe's vision for computer science is that he did not regard the presence of computer science in "the school of letters and sciences," as anything but a step towards a grander goal he had for the discipline. Forsythe stated quite explicitly that he believed that in the future, the organizational structure of universities would change and that CS would have a new sort of home:

> In years to come we may expect a Computer Science Department to mix with departments of pure mathematics, operations research, statistics, applied mathematics, and so on, inside a school of mathematical sciences. We can hope for some weakening of the autonomy of individual departments, and a concomitant strengthening of the ability of a university to found and carry out interdisciplinary programs.

55. G. E. Forsythe. 1968. What to do till the computer scientist comes? *Am. Math. Month.*, 75(5): 456–457.

56. G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 6.

Forsythe singled out the Division of Mathematical Sciences at Purdue[57] as an exemplar of what computer science *could be* if universities were reorganized along the lines he found ideal. However, he conceded the reality was that such reorganizations were far off and that computer science was best served by functioning as a department alongside those found in the sciences, arts, and humanities.[58] Stanford's own Computer Science Department did not follow the trajectory Forsythe had plotted. It remained in the School of Humanities and Sciences until 1985, when it moved to the School of Engineering.[59]

Although Stanford's was not the first computer science department, under Forsythe's leadership it created a widely emulated template for how the field's departments would relate to other disciplines and how they would manage research funding. Like others that followed it, Stanford's Computer Science Department grew out of the Mathematics Department, and would for many decades orient itself more towards mathematics than engineering. Within the department, faculty members were granted considerable freedom and were encouraged to collaborate across disciplinary lines. Pursuing this policy, Stanford computer scientists formed "dukedoms," united only by teaching concerns, in which they pursued their interests largely insulated from departmental politics as well as each other's sometimes contradictory agendas—a pattern replicated widely in many of today's leading departments.[60] As part of seeking to establish computer science as a fully recognized academic discipline, Forsythe encouraged faculty to seek federal research grants to support their work, just as a physicist or biologist would.[61] He encouraged universities to form computer science departments along the lines of the one at Stanford:

57. R. L. Pyle. 2015. *First in the Field: Breaking Ground in Computer Science at Purdue University*. Purdue University Press, West Lafayette, IN. 2015) provides a beautifully illustrated historical overview of early computer science at Purdue.

58. G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 6. In laying out such a plan for computer science, Forsythe's plan resembled Fein's vision of Synnoetics, but stripped of its more grandiose claims.

59. Stanford Computer Science Department timeline. Available at http://www-cs.stanford.edu/about/department-timeline (accessed February 2016).

60. Oral history interview with Edward Feigenbaum. Charles Babbage Institute, OH 14. Available at http://purl.umn.edu/107283.

61. G. E. Forsythe. June 25, 1965. Stanford University's program in computer science. Technical Report CS26, Computer Science Department, Stanford University. Available at http://i.stanford.edu/pub/cstr/reports/cs/tr/65/26/CS-TR-65-26.pdf.

closely associated with mathematics, unhesitant to blend theory and experimentation, and actively seeking to deliver the products of their federally sponsored research to the commercial market.

Ironically, given today's association of Stanford Computer Science with corporate start-up culture (e.g., Google, Snapchat, Coursera), Forsythe's effort to inculcate fast-paced innovation among computer scientists meant using corporate models as a negative example. Pointing to electrical public utility companies, Forsythe admonished computer scientists *not* to embrace the then-popular notion of a computing utility. Granted, there seemed to be viable commercial opportunities connected to running a computing utility, but Forsythe characterized the business model of utilities as "making changes very carefully and slowly, after long periods of tryout and customer research." Such conservatism and sluggishness was incompatible with the act of pioneering use of a rapidly changing technology. He warned: "I conclude that computer folk must either begin to act like utility folk fairly soon, accepting the pain of so carefully weighing each change, or else we should stop our great talk of the soon-to-be-here computing utility."[62]

Another characteristic of the Stanford Computer Science Department that seems to have been widely emulated was its marginalization of women. In recent years, historians of computing have made the case that the heavily disproportionate presence of men in the profession of programming and in the discipline of computer science is the consequence of deliberate actions—on the individual, institutional, and societal levels—to exclude women from computing.[63] Pointing to the overwhelmingly female composition of the very earliest programmers, such as those who operated ENIAC in the late 1940s, Nathan Ensmenger suggests that as programming professionalized, women were pushed out of the field.[64] In a sense, as computing professionalized, it also masculinized. Michael Mahoney, in "Boys' Toys and Women's Work: Feminism Engages Software," argued that widespread adherence to mid-century social norms marginalized women from computing, an

62. G. E. Forsythe. 1965. President's letter. *Commun. ACM*, 8(9): 541.

63. T. J. Misa, editor. 2010. *Gender Codes: Why Women are Leaving Computing*. Wiley IEEE Computer Society, Hoboken, NJ; J. Abbate. 2012. *Recoding Gender: Women's Changing Participation in Computing*. MIT Press, Cambridge, MA.

64. N. Ensmenger. 2010. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*. MIT Press, Cambridge, MA.

activity that they had pioneered, once it became evident that how important computing would become.[65] In this light, one of the important exemplary qualities of Stanford's Computer Science Department, besides its structure and priorities, was its longtime exclusion of women. Not a single woman was tenured there during the first 30 years of the department's existence.[66] Although Forsythe was acutely aware that women could make important contributions to computer, he appears to have done little to challenge prevailing gender norms.

For Diana Forsythe, the treatment of women at Stanford's Computer Science Department was a subject she sought to investigate as a cultural anthropologist. Using her parents' careers as a case study, she illustrated how societal expectations about gender roles created two very different paths for two people with similar backgrounds and capabilities:

> Imagine two bright people, let's call them George and Sandra. In the late 1930s they graduated from Swarthmore College and went to Brown University to do graduate work in mathematics. George encountered supportive faculty and finished his Ph.D. in four years. Sandra, in contrast, found that her advisor did not approve of female mathematicians. She eventually left Brown and completed a master's degree in mathematics at Smith.
>
> Cut to the late 1960s. George is now a full professor at a major university on the West Coast. An applied mathematician, he is helping to develop the new field of computer science and has become internationally known. He has just attained his goal of persuading the university to start a new computer science department, of which he is to be chair. In this position he assembles a computer science faculty, which by the way contains no women. (In the succeeding 25 years, this department produces female Ph.D.'s but hires very few and does not tenure one until 1995 or 1996.) Sandra has also become a computer scientist, although without an academic base. She would like a position at the university but the nepotism rule works against her. So she turns to secondary education and teaches high school mathematics. There she accomplishes what George did in

65. M. S. Mahoney. 2001. Boys' toys and women's work: Feminism engages software. In A. N. H. Creager, E. Lunbeck, and L. Schiebinger, editors. *Feminism in Twentieth Century Science, Technology and Medicine*, pp. 169–185. University of Chicago Press, Chicago.

66. D. E. Forsythe. 2001. Disappearing women in the social world of computing. In D. J. Hess, editor. *Studying Those Who Study Us: An Anthropologist in the World of Artificial Intelligence*, pp. 163–182. Stanford University Press, Stanford, CA.

> the university: she fights for and eventually succeeds in introducing a computer science curriculum, one of the first in the country. She is a prolific author.[67]

Among Alexandra "Sandra" Forsythe's writing on computer science was the popular textbook, *Computer Science: A First Course* (John Wiley and Sons, 1969), several other introductory books to programming languages, and books related to advanced programming concepts. Like her husband, Sandra Forsythe had learned to program on the SWAC, and like her husband she had a strong background in advanced mathematics as well as a liberal arts undergraduate education at Swarthmore. Further like her husband, she had as a child and as an adult been a member of the Society of Friends (Quakers), and both consciously "followed the Quaker injunction to 'speak truth to power.' "[68] Unlike her husband, though, she found that many paths were closed to her, particularly in academics.[69] A "straight A" mathematics graduate student at Brown, like George Forsythe, Alexandra Forsythe had lost her funding because the Dean, "a very conservative man," cut off her "fellowship because he didn't think I properly stayed on the sidelines like a woman should do." Although Alexandra Forsythe cast her mistreatment at Brown as an impetus for the couple to move to the West Coast in the 1940s, she faced similar prejudices there.[70] In the face of such discrimination, her husband and his colleagues seem to have done little to help.

67. Ibid., 168. This book contains sophisticated but incomplete ethnographic studies of Stanford Computer Science. Diana Forsythe's project was cut short by her sudden death at age 50 due to a hiking accident. The book was edited by David Hess and published posthumously.

68. Ibid. It should be noted that the Forsythes' sense of social responsibility did not manifest in public criticism of the powers that be or in activities that authorities viewed as subversive; this is in stark contrast to the experiences of ACM founder Edmund Berkeley, as described in: B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals*. Morgan & Claypool/ACM, San Francisco. George Forsythe did, however, serve on the advisory board for Berkeley's journal *Computers and Automation*. Also on the board was Victor Paschkis, whose efforts to introduce Quaker ethics into the profession of engineering are recounted in: M. Wisnioski. 2012. *Engineers for Change: Competing Visions of Technology in 1960s America*, pp. 1, 72–74. MIT Press, Cambridge, MA.

69. Oral history interview with Alexandra Forsythe. Charles Babbage Institute, OH 17.

70. D. E. Forsythe. 2001. George and Sandra's daughter, Robert's girl: Doing ethnographic research in my parents' field. In D. J. Hess, editor. *Studying Those Who Study Us: An Anthropologist in the World of Artificial Intelligence*, p. 191. Stanford University Press, Stanford.

3.6 Conclusion

George Forsythe did not invent the term "computer science"—that claim may well belong to Louis Fein.[71] And he was not the founder of the first computer science department; there were a half-dozen established before 1965.[72] What he did do, however, was to define computer science in a way that a wide audience could see both its potential and its feasibility. Today, many of his recommendations, such as creating mechanisms to encourage and reward the development of algorithms, seem to be the epitome of obvious sensibility, but there was a time when the correctness of such propositions was far from self-evident. In carefully and respectfully addressing the many seemingly quite valid arguments against investing in computer science as a discipline, Forsythe provided readers of *Communications* the language as well as the intellectual and institutional frameworks they needed to secure a place for computer science.

Forsythe's writings about the formative years of academic computer science suggest that perseverance was an essential component of his success. Many of his contemporaries also saw the merit of establishing computer science, but few so doggedly persisted in the face of well-reasoned skepticism and cruel setbacks. His work also shows the value of his refusal to take the importance of computing for granted. In our world, saturated as it is by computing, it is easy to lose sight of the effort that was required to create a space for people to think systematically about how to work productively with information.

71. A. Leonard. June 10, 1999. "Can history survive Silicon Valley?" *Salon*. Available at http://www.salon.com/1999/06/10/stanford/.

72. G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 4.

4

Solving a Career Equation: The First Doctoral Women in Computer Science

Irina Nikivincze

4.1 Introduction

During the rapid growth of computer science as a discipline in the 1960s and the 1970s, women were entering the fields of science, engineering, and mathematics in increasing numbers. In the late 1960s, about 11% of all bachelor's degrees in computer science were awarded to women, but only a handful of women pursued doctoral-level training.[1] A survey of 60 Ph.D.-granting computer science departments in the U.S. reported that the percentage of doctorates received by women had increased sharply in the 1970s, reaching nearly 10% by 1976. By earning advanced degrees in a technical field, these women were challenging gender stereotypes and literally "inventing [their] careers."[2] This chapter examines the career patterns of the first women to receive doctoral degrees in computer science in the U.S. between 1970 and 1976. These pioneers comprise a distinguished group of researchers who followed their interests, entered a new field, and persisted in computer science in both academia and industry while remaining largely invisible to the public.

1. In the 1960s, women earned less than 3% of all doctoral degrees in computer science. See Figure I.1 in J. McGrath Cohoon and W. Aspray, editors. 2006. *Women and Information Technology: Research on Underrepresentation*, p. x. MIT Press, Cambridge, MA.

2. O. E. Taulbee and S. D. Conte. 1977. Production and employment of Ph.D.'s in computer science—1976. *Commun. ACM*, 20(6): 370–372; J. Abbate. 2012. *Recoding Gender: Women's Changing Participation in Computing*, p. 4. MIT Press, Cambridge, MA.

Historians of science and computing often explain the invisibility of women and their contributions by cultural and structural factors such as their traditionally low status in the computer industry (e.g., as punch card operators), gendered assumptions about skills and jobs, stereotypes about computer-related work, and the general devaluation of women's work. Women were present in significant numbers in the early decades of computers as calculators, programmers, keypunch operators, data processors, managers, and entrepreneurs.[3] Although educational opportunities for women expanded and many pursued advanced degrees, the number of women entering physical sciences, mathematics, and engineering remained low.[4] To enter computing as technical professionals rather than keypunch operators, women had to redefine women's work. Becoming a computer professional meant being "one with the guys" with all that it entailed: membership in the Association for Computing Machinery (ACM) as opposed to the Data Processing Management Association; attendance of technical presentations at the National Conference of the ACM as opposed to registration for "Women's Activities" or SIGWIVES with trips to farmer's markets or Disneyland.

Surprisingly, little is known about why women chose to pursue careers in the science of computing. We are only able to shed some light on the lives and the careers of women who followed their interests and became research scientists—those who succeeded despite the traditional gender expectations of their time. By tracking the advances of women from a historical perspective, we uncover patterns of participation and contributions to the emerging field of computer science. This exploration reveals the factors that supported women's early computing careers and the strategies that could help attract women to computing in the future. Although statistics provide evidence of the scope of women's participation in the computing field, we must search beyond the numbers to fully comprehend their experience.[5] Therefore, this study uses ten interviews: seven collected with the support of the

---

3. J. Light. 1999. When computers were women. *Technol. Cult.*, 40(3): 455–483, available at http://www.jstor.org/stable/25147356; D. A. Grier. 2005. *When Computers were Human*. Princeton University Press, Princeton, NJ; Abbate, *Recoding Gender*; C. Schlombs. 2010. A gendered job carousel: Employment effects of computer automation. In T. J. Misa, editor. *Gender Codes: Why Women are Leaving Computing*, pp. 75–94. Wiley IEEE Computer Society, Hoboken, NJ.

4. T. A. DiPrete and C. Buchmann, 2013. *The Rise of Women: The Growing Gender Gap in Education and What it Means for American Schools*. Russell Sage Foundation, New York.

5. C. Hollenshead, S. Wenzel, B. Lazarus, and I. Nair. 1996. The graduate experiences in the sciences and engineering: Rethinking a gendered institution. In C.-S. Davis, et al., editors. *The Equity Equation: Fostering the Advancement of Women in the Sciences, Mathematics, and Engineering*, pp. 122–163, on 128. Jossey-Bass Publishers, San Francisco.

Association for Computing Machinery History Committee, and three oral histories collected by Janet Abbate. The interviews explore various stages of women's professional work: graduate training, career paths, work, recognition, gender issues, and strategies for success. Conclusions drawn from the interviews are supplemented by articles from *Communications of the ACM*, *Datamation,* and *Computerworld*.

The chapter begins by laying a historical foundation of the social, political, and economic factors influencing computer science and exploring gender dynamics in science and computing in the 1970s. It continues by describing the data and the research strategies used in building a cohort of women computer scientists and then reporting on the findings of an aggregated analysis of their careers in what can be called a "collective biography" of the cohort. The chapter ends with a summary of findings and the conclusion.

4.2 Historical Context

Before computing could develop into "an independent new science," it had to be accepted as a legitimate discipline. The development of computer science was a long process muddled by an uncertain identity of the new area of inquiry.[6] The work associated with early computers was performed by employees with a range of educational backgrounds in mathematics, physics, and engineering. A number of well-known female computer scientists such as Grace Hopper, Mina Rees, and Evelyn Boyd Granville were trained as mathematicians, others such as Thelma Estrin and Martha Sloan as engineers, and others such as Gwen Bell as geographers. The appearance of training and degrees defined and launched a lengthy process of solidifying the professional identity of computer professionals.

Computing first found its place in academic departments of engineering and mathematics and in the 1960s made its claims as an independent science. However, the earliest computer departments in the U.S., according to Edsger Dijkstra, "were no more than ill-considered cocktails of presumably computer-related topics that happened to be available on campus."[7] Although Purdue University established the first department of computer science in the U.S. in 1962, it was not until 1965 that preliminary curricula published by the Association for Computing Machinery

6. J. Hartmanis. 1994. On computational complexity and the nature of computer science. *Commun. ACM*, 37(10): 37–43, quote on 41; S. V. Pollack, editor. 1982. *Studies in Computer Science*. Mathematical Association of America, Washington, DC.

7. E. W. Dijkstra. 1986. EWD952: Science fiction and science reality in computing, 1986. Available at http://www.cs.utexas.edu/users/EWD/ewd09xx/EWD952.PDF.

provided "a coherent definition" of the discipline.[8] In the late 1960s, other universities formed independent units that began granting degrees in "computer science." By 1967, at least 29 universities offered graduate curricula in computer science in various departments and under various titles.[9]

About 11–15% of those receiving bachelor's degrees in computer science from 1966–1974 were women. During the 1960s, the National Science Foundation reported that there were only three women among 157 doctoral degree graduates. Among the three was Sister Mary Kenneth Keller. In 1965, at the age of 52, she became the first woman to receive a Ph.D. in computer science for her dissertation, titled "Inductive inference on computer generated patterns" from the newly formed Department of Computer Sciences at the University of Wisconsin–Madison.[10] Although some believed that university computer science departments "have been bending over backwards" when it came to admitting women and minorities, from 1971–1975 only about 15% of master's degrees recipients and 7% of doctoral degree recipients were women.[11] Low numbers of women in computer science programs in the 1960s and 1970s mirrored a larger trend of their low participation in other fields of science.

4.3 Gender and Science

The period from the mid-1950s to the late 1960s was "one of tremendous growth and prosperity for science at American educational institutions." With the help of federal funding, colleges and universities created new research laboratories, built new libraries, purchased new equipment, and expanded their faculties. However, historian Margaret Rossiter found that when it came to academic rank and employment, women scientists did not fully benefit. In the 1960s, women scientists were few, isolated in their disciplines, and marginal in rank and power. Many universi-

8. Pollack, *Studies in Computer Science*, 35.

9. Colleges and universities reporting "computer science" curriculum. August 29, 1967. Charles Babbage Institute (CBI 205), Association for Computing Machinery Records Box 25, Folder 34.

10. L. Thurgood, M. J. Golladay, and S. T. Hill. 2006. *U.S. Doctorates in the 20th Century*. National Science Foundation, Arlington, VA. Keller's Ph.D. was verified by staff at the University of Wisconsin library. The ProQuest database classified her dissertation as mathematics.

11. P. J. Denning. 1977. Discrimination and affirmative action. *Commun. ACM*, 20(3): 198; R. Montanelli and S. Mamrak. 1976. The status of women and minorities in academic computer science. *Commun. ACM*, 19(10): 578–581.

ties had anti-nepotism laws that prohibited the hiring of spouses, creating a major problem for dual-career couples.[12]

The result of the tremendous expansion of universities and laboratories from the late 1950s to early 1970s—the "golden age" for science in America—was in fact a "very dark age" for academic women. Few women were faculty members at the 20 leading research universities. Although their numbers were higher in departments of home economics, genetics, anatomy, psychology, physiology, and bacteriology, women constituted no more than five percent of all faculty in the remaining 16 science fields. Even having one woman (a practice commonly referred to as "tokenism") "would have been quite an advance from the norm at prestigious universities at the time." Instead of academia, women scientists tended to opt for positions in industry, nonprofit organizations, or the federal government, or for entry-level jobs in technical fields. Even in these positions, however, women faced "the start of a career-long struggle with a succession of gender issues."[13]

In the second half of the 20th century, science remained an institution with "immense inequality in career attainment." Women often occupied the margins of the scientific community when it came to visibility, institutional location, and status. As newcomers, many stated that they were treated differently. Prominent scientists such as Evelyn Fox Keller, Barbara McClintock, Salome Waelsch, Andrea Dupree, and Sandra Panem often considered themselves to be outsiders and even "deviants" in the world of science.[14] By venturing into science, these women encountered estrangement, ambivalence toward their work, and invisibility. The persistence of such experiences prompted Margaret Rossiter in 1993 to name the phenomenon of invisibility and the recurring denial of credit for achievements of women the "Matilda effect." Unfortunately, this situation was not much different for women in academic computing.[15]

---

12. M. W. Rossiter. 1995. *Women Scientists in America: Before Affirmative Action, 1940–1972*, p. 122. Johns Hopkins University Press, Baltimore.

13. Ibid., 123, 129, 142.

14. J. S. Long and M. F. Fox. 1995. Scientific careers: Universalism and particularism. *Ann. Rev. Sociol.*, 21:45–71, quote on 45; E. F. Keller, 1985. *Reflections on Gender and Science*. Yale University Press, New Haven, CT; H. Zuckerman, J. R. Cole, and J. T. Bruer, editors. 1992. *The Outer Circle: Women in the Scientific Community*. Yale University Press, New Haven, CT.

15. Rossiter documented the under-recognition and invisibility of women in science and in history in general and suggested that this phenomenon be called the "Matilda effect" in memory of Matilda J. Cage, a late 19th century American suffragist who became aware of and denounced a widespread practice of the denial of credit to women. See M. W. Rossiter. 1993. The ~~Matthew~~

4.4 Gender and Computing: Identifying the Problems

In the mid 1960s, legislative pressure and affirmative action promoting equal access to educational opportunities provided women with greater opportunities in education and the workforce. The Equal Pay Act of 1963 banned sex discrimination in pay, and the Civil Rights Act of 1964 prohibited employment discrimination on the basis of race, color, religion, sex, or national origin. The Higher Education Act (HEA) of 1965 and the Education Amendments of 1972, specifically the Title IX clause, which prohibited sex discrimination at academic institutions receiving federal aid, opened the doors to women's participation in higher education.

These initiatives strengthened women's leadership and participation in computing in the 1960s. U.S. Navy admiral and COBOL "inventor" Grace Murray Hopper received the first "Computer Sciences Man-of-the-Year" award in 1969 from the Data Processing Management Association (DPMA). Jean E. Sammet, an IBM employee involved in the ACM, sought a "legitimate forum" in which academic and industry researchers could exchange information about FORMAC (FORmula Manipulation Compiler) and symbolic/algebraic computation (numerical analysis already had a special interest group at ACM). She set up such a forum by first creating a special interest committee (SIC) and organizing the Symposium on Symbolic and Algebraic Manipulation (SYMSAM). She later became a Northeast regional representative, joined the ACM Council, and advanced to other leadership positions, including chair of the ACM Committee on Special Interest Groups and Special Interest Committees, chair of the ACM Special Interest Group on Programming Languages, ACM vice president from 1972–1974, and ACM president (the first woman) from 1974–1976. "An independent-minded soul," Sammet shaped the history of computing by supporting programming languages, developing SIGs, rescuing ACM from financial straits, promoting the ACM Fellow Program, and improving the overall organization of the ACM.[16]

Despite the advances of some women in computer science, concerns for the status of women in computing and discrimination in the workplace were repeatedly voiced in the pages of computer magazines, specifically the flagship publication *Communications of the ACM*. "Discrimination in the Employment of Women in the Computer Industry" concluded that women were paid less and "[did] not yet have equality with men." Other studies reported mixed results. In 1975 *Datamation* pub-

Matilda effect in science. *Soc. Stud. Sci.*, 23(2): 325–341; see M. Klawe and N. Leveson. 1995. Women in computing: Where are we now? *Commun. ACM*, 38(1): 29–35.

16. T. J. (Tim) Bergin. 2006. Jean E. Sammet Interview: March 28, April 4, April 11 and April 18, 2006. ACM Oral History Interviews, quotes pp. 1, 13. ACM, New York. DOI: 10.1145/1141880.1243440.

lished a survey of women in data processing that revealed the equal status of women with their male colleagues in pay but limited opportunities for advancement to top managerial positions.[17] In that same year, *Computerworld* reported that a majority of women data processors had not encountered job discrimination, but in the words of one woman, "men find it difficult to work with a woman no matter where you are." Although a few years earlier, a woman in computing would have been frequently asked, "What [am] I doing in a man's field?"[18] by the mid-1970s women generally felt that their circumstances had improved and that the new field was fair and open to women. The feminist movement had improved the situation, but by how much? How well did women researchers fare in the computer field?

The career experiences of academic women in computing have been brought to light by Janet Abbate's 2012 book *Recoding Gender*. In her study, Abbate collected 52 oral history interviews with women occupying various computing positions in industry and academia in the U.S. and the U.K. Her analysis of gender in academic computing revealed that some women followed non-traditional paths to success in academia. Some held academic jobs between 1965 and 1979 while holding only bachelor's or master's degrees. Others succeeded by receiving support and visibility through active work in professional societies or the establishment of computer science forums specifically for women. This study examines the careers of the first women in the U.S. who followed the traditional route to academia: entering graduate school and earning doctorates in computer science.

4.5 Data and Method

National Science Foundation statistics on the number of degree holders by gender served as a guide for identifying the number of women who earned computer science degrees from the 1960s until the mid-1970s.[19] The ProQuest *Dissertations and Theses* database contains the names of the graduates who received a doctorate in computer science. It was used to identify the first women (N=93) to have received dissertations in computer science in the U.S. between 1970 and 1976 by

17. R. Weber and B. Gilchrist. 1975. Discrimination in the employment of women in the computer industry. *Commun. ACM*, 18(7): 416–418; W. Asprey and W. Laffan. 1975. Women speak out on DP careers. *Datamation*, 21(8): 41–44.

18. C. Arnst and A. Dooley. July 23, 1975. Majority of women DPers find no job discrimination. *Computerworld*, 1–4, quotes on p. 4.

19. L. Thurgood, M. J. Golladay, and S. T. Hill. 2006. *U.S. doctorates in the 20th century.* National Science Foundation, Arlington VA.

authors' first and middle names.[20] Only one-third (N=30) of the identified women computer scientists with a Ph.D. had both published biographical information[21] and work experience in academia, so they were included in the sample.[22] For the other two-thirds of the women graduates, Internet searches provided clues about their careers. Some had become industry professionals, programmers, software engineers, or corporate executives. A few had become writers, and one had switched to psychology. However, for many others (38% of the 93), no biographical or professional information was available in online sources (not even a professional contact), so I was not able to determine if they had left computer science or stayed in the field.[23]

This chapter reports on an analysis of interviews with women computer scientists, an extraordinary group driven by their passion for the subject and the desire to succeed. From a sample of 30 identified women computer scientists with doctorates, I chose 10.[24] After the ten were contacted, seven were interviewed (five in person and two electronically): Lori Clarke, Dana Angluin, Dana Ulery, Anne-Louise Radimsky, Mary Shaw, Sandra Mamrak, and Dianne O'Leary. The other three women, Adele Goldberg, Ruzena Bajcsy, and Susan Graham, had already been interviewed on multiple occasions, so the transcripts of these interviews were used in the study. The interviews focused on graduate and work experiences, careers, achievements, gender, challenges, and strategies for career advancement. All of these women shared their personal and professional life stories. The interviews were transcribed, imported, and thematically coded in ATLAS.ti. In this chapter I present their individual voices as well as similarities, a kind of a collective bio-

20. No names were found for the 1960s.

21. Published biographic information was located by using directories such as *American Men and Women of Science,* first or last pages of their dissertations, "people search" engines, LinkedIn, and by collecting online curriculum vitas.

22. The selected one-third (N=30) of identified women represented a group of researchers who had persisted in computer science and who had either an academic career or a mixed career combining industry and academic experience. Many of these women Ph.D. holders between 1970 and 1976 had long careers, some had just recently retired, while others were still working. The occupational attainments of these women were analyzed and compared with those of Turing Award winners; see I. Nikiforova. 2012. The paradox of excellence: Merit and occupational attainments of women in computer science. *Proceedings from GEXcel Theme 11–12: Gender Paradoxes in Changing Academic and Scientific Organisation(s)*, pp. 49—56. Linköping University Press, Linköping: LiU-Tryck.

23. Although most women were married by the time they graduated with a Ph.D., some of the women may have married later on and changed their last names.

24. Some were selected for specific characteristics (institution, career track); the availability of contact information, however, also played a role.

graphic portrait.[25] To protect their privacy, individual names are not used in this chapter's citations.

4.6 Findings

4.6.1 Graduate Schools

Most of the women included in this study had graduated from top computer science programs. Out of ten cases, six had graduated from the top five[26] computer science departments: three women from Stanford, two from the University of California-Berkeley, and one from Carnegie Mellon University. They had graduated from computer science programs with a variety of names, including engineering science, electrical engineering and computer science, and information sciences. Table 4.1 lists the schools and the titles of the women's doctoral dissertations.

4.6.2 Career Choice: Why on earth do you want a Ph.D.?

While in graduate school, these women often faced the question from their professors: "Why on earth do you want a Ph.D.?" In the 1960s and 1970s, many still considered a woman's only choice was to become "a secretary, a nurse, a teacher" or a flight attendant. None of the women interviewed wanted to become any of these, nor did they want to become professors. Instead, they reported having no expectations and simply "falling into it [academic computer science]." Some admitted to being uncertain about their decision and not knowing whether it was "the right thing to do." One expressed that "Every step of the way it was sort of see if you like it," and then stay or go. They all stayed.

These women were entering an exciting field in which the rules had not yet been defined. Computer careers, however, still demanded that students meet one prerequisite: a strong background in mathematics. Describing their experiences as students, they typically commented that they had always loved math—that it was

25. Historians sometimes refer to an analysis of interviews (qualitative data) as a "collective biography" or "prosopography"; see N. L. Ensmenger. 2001. The "question of professionalism" in the computer fields. *IEEE Ann. Hist. Comput.*, 23(4): 1–19; R. A. Nelsen, Race and computing: The problem of sources, the potential of prosopography, and the lesson of *Ebony* magazine. *IEEE Ann. Hist. Comput.*, (forthcoming 2017). (Updated publication available at http://doi.ieeecomputersociety.org/10.1109/MAHC.2016.11.)

26. The top five departments were identified based on M. L. Goldberger, B. A. Maher, and P. E. Flattau, editors. 1995. *Research Doctorate Programs in the United States: Continuity and Change*. National Academy Press, Washington, DC.

Table 4.1 **Universities, dissertations, and first positions of women scientists (N=10).**

Year	Name	Ph.D. University	Department/Program	Dissertation Title	First Job (after Ph.D.)
1971	Graham, Susan L	Stanford	Computer Science	Precedence Languages and Bounded Right Context Languages	Assistant Prof. at UC Berkeley
1972	Shaw, Mary M	CMU	Computer Science	Language Structures for Contractible Compilers	Assistant Prof. at CMU
1973	Bajcsy, Ruzena K	Stanford	Computer Science	Computer Identification of Textured Visual Scenes	Assistant Prof. at UPenn
1973	Radimsky, Anne-Louise Guichard	UC Berkeley	Electrical Engineering and Computer Sciences	Semantic Analysis of English Text by Computer	Assistant Prof. at UC Davis
1973	Goldberg, Adele J	U of Chicago	Information Sciences	Computer-Assisted Instruction: the Application of Theorem-proving to Adaptive Response Analysis	Researcher at Xerox
1975	Mamrak, Sandra A	U of Illinois	Computer Science	Comparative Response Times of Time-Sharing Systems on the ARPA Network	Assistant Prof. at Ohio University
1976	Clarke, Lori A	U of Colorado	Computer Science	Test Data Generation and Symbolic Execution of Programs as an Aid to Program Validation	Assistant Prof. at UMass
1976	Angluin, Dana Charmian	UC Berkeley	Engineering Science	An Application of the Theory of Computational Complexity to the Study of Inductive Inference	Succession of Postdocs at Leeds, UK; U of Edinburgh; UC Santa Barbara
1976	Ulery, Dana L	U of Delaware	Applied Science	Computer Science's Reincarnation of Finite Differences	Research Scientist at DuPont
1976	O'Leary, Dianne Prost	Stanford	Computer Science	Hybrid Conjugate Gradient Algorithms	Assistant Prof. at U of Michigan

something they had always been good at. They also enjoyed a challenge, so they often took the toughest classes. In a number of instances, the decision to go to graduate school came as a reaction to an environment that was not mentally challenging enough, one in which they felt isolated. One woman said, "I mentally had no challenges, no contact with people who talked about interesting problems . . . I felt wasted . . . so I decided to go back to school."

It was also a time when computers were new. Few of the women knew much about them. Some of them worked around computers and computers piqued their curiosity. They wanted to learn something about which they knew nothing. Since few co-workers understood them, the women were eager to take on the challenge, saying, "I just felt I should know more . . . and if I wanted to be better skilled, I needed to go to graduate school." Not knowing was not only an uncomfortable position, but also a challenge: "I went for the knowledge, feeling I didn't know enough"; "it's another vocabulary, . . . and I felt I needed to know it." Thus, rather than shy away from learning, these women saw an opportunity to challenge themselves and explore computers that they perceived to be a *fun* activity. The excitement of their interactions with computers was revealed in the way women spoke about their experiences in the field—about having "fun playing with computers." Although they had no "grand plan" of what they were going to do with their new knowledge, in many cases, the women, who had begun by pursing master's degrees, eventually earned Ph.D.'s.

4.6.3 Advisors

In the eyes of the interviewees, advisors made a world of difference in their careers. Women described their advisors as enthusiastic, encouraging, fun to work with, patient, and never disappointed. Advisors shared their ideas with students and created a collegial and creative environment for all students, regardless of gender. The influence of advisors was clear in the following comments: "Alan Perlis was a reason for joining CMU," "Hatem Khalil's encouragement and confidence was a major factor in my decision to pursue this degree." Women stressed the influence of such interactions, one saying, "If I had not had these men [dissertation advisors], I would not have stayed in the field." In one case, however, the indifference of an advisor led to her failing the comprehensive exam, an experience that she was never able to forget. Only by switching her advisor did she continue to finish her Ph.D. and remain in academia.

4.6.4 First Jobs

After graduation, most of the interviewees transitioned to academic jobs as assistant professors, and only two entered industry. The academic positions, however, in most cases were not at top universities in the field. The women often described their early work places (academic departments or industry) as lonely places where most colleagues were indifferent or even overtly biased. One woman, describing the available workplaces, stated that departments typically operated in an "attack, defend style," which was supposed to be "good for discussion and research." As she did not find such an environment pleasant, she chose to work at a more collegial and collaborative school, which, it turned out, was an excellent career choice, for she became quite successful, eventually getting tenure at an institution where only 10% of faculty gets it. Although most of the women recounted difficult experiences, a few landed positions in more supportive academic work places, which created a far different experience for these fortunate women who remained in their departments and were more productive.

Women often reported being the first and, in some cases, the only woman to work in a particular department or laboratory. In some cases, no other women were hired for the next two decades. Not surprisingly, the slow or non-existent progression of women entering the male-dominated domains perpetuated gender expectations of the time, which impeded work relationships among men and women. For example, men did not appear to know how to treat women as colleagues rather than dates. They felt uncomfortable when women paid for lunch. They believed that if they invited a woman to lunch, the man had to pay, so they avoided such invitations. In addition, pay scales were inequitable. Two of the women that were paid less for the same work and the same rank were understandably upset about the inequitable treatment.

Some may argue that these women scientists chose careers that in some way conformed to a gender stereotype. After all, many entered academic institutions to *teach* either math or computer/software courses. In addition, women scientists typically chose "female-friendly" topics related to computer education, software, computer languages, or health. Nevertheless, one could view these choices from a slightly different perspective. They demonstrate that women exercised their preferences and personal judgment by gravitating to research topics with the greatest impact, topics that motivated them to remain in the field and contribute to it. In other words, the women were strategic in the application of their knowledge in society.

4.6.5 Careers in Computing

In most cases, the women's academic careers progressed as expected (see Table 4.2), from assistant to associate and from associate to full professor. For some, on the other hand, the progress was haphazard: leaving academia to have children and then returning, switching from academia to industry, or vice versa. Overall, however, the women reported loving research and teaching, and being satisfied about how their careers had progressed, often saying that they had "the best job in the world." Learning was a key part of it. Because the field constantly evolves, their learning never stopped. As one scientist put it, "I've been in the field for 40 years and am never bored." More importantly, overall, they felt as if they had made a difference in the world.

4.6.6 Changes in Computing Over Time

The interviewees agreed that significant changes had taken place in the computing field since the 1970s. Still working (or recently retired), several were in unique positions to compare the past to the present. They acknowledged that although computing remains competitive, some aspects have become easier, while others have become more difficult. For example, maternity leave is more common, so managing parenting has become easier, but securing funding and sustaining research programs became much more difficult: "Research will continue where resources are. Google has a lot of resources that we cannot build in universities. Talent will end up where resources are. Research will not be centered in universities, . . . a scary development."

The women were not very concerned about locating the core and the boundaries of computer science nor did it seem to impact their research. "I've heard every area of computer science other than operating systems and programming languages," noted one interviewee, "denigrated as 'non-core' by some expert at some time over the past 30 years, including theory of computation, databases, and scientific computing. This does not lead to productive discussions. Definitions of the core naturally evolve over time. In some sense, it is like an apple: the core is only useful in relation to the fruit that surrounds it and the future fruit that it enables." The other scientist remarked that it was "irrelevant" as "disciplines are porous and the divisions are artificial." Yet, they believed that it was important for computing to remain inclusive: "I think computer science is the strongest when it takes this big tent philosophy, being inclusive rather than chopping and dicing down to an artificial core."

The women believed that computing had greatly influenced other disciplines by introducing a computational way of thinking about all subjects, but they did not expound on how the computational methodology influences gender balance in scientific fields. Although many women stated that if faced with a choice of fields to study, they would select computing again, but one woman felt that computing was no longer that attractive as a field to enter, saying, "I don't think I'd go into

Table 4.2 **Places of employment, affiliations, and research areas of women scientists (N=10).**

Year	Name	Major Employment	Memberships/Fellowships	Research Area
1971	Graham, Susan L	UC Berkeley	fel ACM; fel IEEE; fel AAAS; NAE	Software and Engineering
1972	Shaw, Mary M	CMU	fel ACM; IEEE-CS; fel IEEE; fel AAAS; NY AS; Sigma Xi	Software and Engineering; Computer Science Education
1973	Bajcsy, Ruzena K	UPenn	fel ACM; fel IEEE; fel AAAS; fel AAAI; NAE; fel NAS	Artificial Intelligence
1973	Radimsky, Anne-Louise Guichard	California State University, Sacramento	ACM; IEEE; SWE, Sigma Xi	Computer Science Education
1973	Goldberg, Adele J	ParcPlace-Digitalk; Neometron; AgileMind; Bullitics [a]	fel ACM	Software and Engineering; Management
1975	Mamrak, Sandra A	Ohio University	ACM	Computer Systems Organization; Information Systems
1976	Clarke, Lori A	UMass	fel ACM; fel IEEE; IEEE-CS	Software and Engineering
1976	Angluin, Dana Charmian	Yale	ACM [a]; SIAM [a]; ACL [a]; AAAI [a]	Theory of Computation
1976	Ulery, Dana L	DuPont; US Army Res Lab	ACM; IEEE-CS; AAAS; Sigma Xi	Information Systems
1976	O'Leary, Dianne Prost	University of Maryland	fel ACM; fel SIAM; AWM; AAAS	Mathematics of Computing

a. *Not confirmed*

computing. I think I'd probably be more interested in biotech, something that has a higher impact and an interesting mixture of what you can accomplish. You'd still use computers, but . . . now it's more that computers are a basis for computation—but compute what? . . . I wonder if that isn't the reason why you don't get as many women in computer science today."

Looking back, the interviewed women wished to change a number of things about academic computer science. First of all, computing needs more funding. Second, departments need to remove uncertainty from tenure processes through clarity, transparency, and mentoring. Third, universities should stop the tenure clock when a tenure candidate is expecting a child. Finally, "something needs to be done about those conference publications." The publications, often of poor quality, may be hurting the discipline: "I would like computing researchers to have more respect for their work by emphasizing archival literature. The emphasis on research driven by conference deadlines, rushed reviews, and rushed revisions means that publications are too often premature and without broad perspective. This mode of publication is not common in the sciences, and it encourages a lack of respect for our work from colleagues in the physical and biological sciences."

4.6.7 Scientific Communities

Like other women in science, the women computer scientists began their careers somewhere between the "outer" and "inner" circles of science. Seeing where they found themselves at the end of their careers and their relationships to scientific communities was of great interest to this study. All were members of the ACM community. From their membership came their identity as computer scientists. At least three women were actively involved in ACM activities: Mary Shaw re-wrote the ACM Computing Classification System for her research specialty, Lori Clarke was actively involved with ACM-SIGs on programming languages and software engineering, and Adele Goldberg served as ACM president from 1984 to 1986. Dana Angluin co-founded the Computational Learning Theory (COLT) Conference sponsored by the Association for Computational Learning (ACL). Six women had received appreciation from the ACM for their contributions to the field and became ACM Fellows.

Women scientists had a strong professional identity. Although the choice of a professional society depended on one's research interests (Table 4.2),it was also influenced by graduate training and the composition of their graduate departments. Several women retained close ties to engineering (and became members of IEEE,

IEEE-CS, and NAE), others with the mathematics of computing (and became members of the Society for Industrial and Applied Mathematics (SIAM) and ACL), and yet others identified mainly with computer science and the ACM.

The women also differed in their collaborative strategies. Some "cosmopolitans" interacted primarily with professional networks such as the Society for Industrial and Applied Mathematics (SIAM) and the ACM, while other "locals" collaborated within or close to institutions such as surrounding area hospitals, businesses, or research centers. Unlike those who associated themselves primarily with a campus, a department, or a university, computer scientists still active in the field often collaborated both locally and nationally.

4.6.8 Recognition

The studied group of women primarily consisted of a distinguished group of researchers with long careers in computing. However, none of them have received a Turing Award.[27] Only three women (not studied) have received the Turing Award: The first was Frances Allen in 2006, then Barbara Liskov in 2008, and Shafi Goldwasser in 2012. Their awards came after many years of professional work (an average of 36 years from their terminal degrees compared to 27 years for their American male counterparts). In light of the above comparisons, one question remains: How did women perceive recognition of their achievements? The general response was that they felt "reasonably recognized" for their work. Some stated simply that they did "work worthy of recognition" but did not dwell on the idea of recognition. Some claimed that nobody feels fully recognized in computer science and that it would be nice to be "re-discovered." Others speculated about how much more recognition they would have received if they were men. Most of them regarded the reward system in computer science as fair, albeit more burdensome for women. For the most part, the women were generally humble about their achievements and appeared to be uncomfortable talking about the subject, which may have revealed a degree of ambivalence toward the subject and their status in the field.

27. A comparative study of occupational attainments of the thirty women in the sample (which included the ten women in this study) revealed that their productivity and citations of *some* of their publications were comparable to those of Turing Award winners; see I. Nikiforova. 2012. The paradox of excellence: Merit and occupational attainments of women in computer science. *Proceedings from GEXcel Theme 11–12: Gender Paradoxes in Changing Academic and Scientific Organisation(s)*, pp. 49—56. Linköping University Press, Linköping: LiU-Tryck.

4.6.9 Gender

None of the interviewees overtly complained about gender having been an obstacle to their careers, but most admitted that, in retrospect, gender had played a role in their careers. They reported that they had indeed been treated more informally. According to one woman, "They called males Dr. Smith, but I was called Sandy." Moreover, visitors and new students had often confused them with support staff, and undergraduate and graduate students had often mistaken women professors for master's degree students. Therefore, in response, the women had developed various strategies to avoid such misunderstandings, such as learning to dress appropriately and to introduce yourself formally.

Women often described subtle ways in which gender played a role in their workplaces. As mentioned before, women sometimes had received less pay for the same work, and they had been evaluated by different standards than their male counterparts. They mentioned having colleagues who did not trust women, colleagues who had problems interacting with women, and colleagues who made off-color remarks. Instances of sexual harassment were also common. One woman stated, "While I experienced sexual harassment 40 years ago as a college intern and as an assistant professor, I never expected it would still be prevalent today. Yet in the last year, I have counseled an undergraduate who was asked by a much older colleague to 'make out' with him in a conference room during her internship, a faculty member who received a lewd message from an undergraduate in her class, and another faculty member who was asked to have sex by a famous colleague who was writing a promotion recommendation for her." Such behavior exemplifies the persistence of gender issues in science. Not surprisingly, avoiding such encounters by de-emphasizing one's femininity was one of the solutions.

4.6.10 Family

An insightful essay written by Judy Syfers in 1971 in *Ms. Magazine* explained why, similar to her divorced male friend, she also wanted a "wife." She wrote that a wife would support her while she attended school, take care of her children, hold an extra job outside of the home, and take care of her well-being and sexual needs while not demanding too much attention. She would also see to it that her clothing, food, cleaning, and health needs would be met. She would manage the household, entertain guests, help with writing papers and allow her to do whatever she wants. While such arrangements have worked well for men, women, relegated to these supporting roles, faced greater obstacles in academia from institutional practices that held them accountable to masculine norms and standards.[28] The irony of this essay perhaps was not lost on a number of women scientists at that time, including those interviewed in this study.

Many of the studied women successfully combined their roles in academia with their roles as wives and mothers (at least four women were married while in graduate school, and one had children). They admitted that balancing the two was not easy, and it required hard choices. Those at the most elite schools opted not to have children or had children prior to starting their doctorates, or left their jobs to care for children or family. Others divorced unsupportive husbands, left childcare to others, or chose positions at less prestigious institutions that would allow them to balance work and family responsibilities. Since they were following non-traditional careers, "working against what [they] did not want to do [to be a secretary, a nurse, or a teacher]," they were considered "rebels." Others were considered "wild." At the same time, most acknowledged that family members, specifically husbands, were a major means of support during their careers. They also received support from colleagues in their departments and the National Science Foundation, both of which enabled them to remain in or re-enter academia to pursue their careers.

Given the persistence of stereotypical gender roles and the demands of academic practices, family responsibilities have always required greater sacrifices from women scientists than from their male counterparts. "I know of a few women and but no men who left academia because they couldn't balance their career and their family," admitted one interviewee. Another explained that since research demands a great deal of time and effort and since men were "used to ignoring family,"

28. J. Syfers. 1971. December 31, 1971. Why I want a wife. *Ms. Magazine*, pp. 3–5. See L. Bailyn. 2003. Academic careers and gender equity: Lessons learned from MIT. *Gender, Work Org.*, 10(2): 137–153 who assessed the "gendered nature" of academic rules—rules that academic scientists take for granted such as the criteria of evaluation, the convention of authorship, a tenure timetable of having to prove oneself in the first seven years (demanding the most productivity during child-bearing years)—favor men and disadvantage women. Academia presupposes an image of an ideal worker as someone "who gives total priority to work and has no outside interests and responsibilities" (p. 139). Being completely gender-neutral, the definition of equality "ignores the different life experiences of men and women and makes the current 'male' model of the ideal academic normative" (p. 139). She argues that an equitable approach "should entail equal opportunities and equal constraints" (p. 140). The idea that present procedures are fair, objective, and gender neutral do not consider uneven constraints that men and women face in their lives. An outcome of such arrangements is that many male professors at institutions such as MIT have families and children, while many women professors do not. The women are either single or divorced and have no children. In other words, academic practices constructed by men, and perceived to be gender neutral, work against women.

they were successful in their careers, but for many women "family always came first" while research "was second." Several of the women admitted that circumstances were finally changing, saying that they were delighted to see "how men are changing and raising children."

4.7 Conclusion

The ten women scientists exhibited several common characteristics: they followed their interests and overcame cultural norms by becoming "technical" computer professionals; they were energetic, organized, and highly motivated; and they had "fun" working in the field. These women were pioneers, creating a path for others to follow. As students, they had strong mathematical skills and a relentless desire for knowledge and learning. Often, they were the only women in their classes and their departments. Many reported having a supportive and enthusiastic advisor whom they credited for their remaining in the doctoral program and continuing their education. Later on, they received support from colleagues, family members, funding agencies, and professional organizations that enabled them to continue their academic careers. Although they did not intend to become professors, they somehow "drifted" into Ph.D. programs and "fell into" careers, which turned out to be highly successful. They were able to secure research funding, support their labs and their research assistants, and publish the results of their research. They had either held one job at one institution or easily found one at another. Reflecting on their careers, all of them considered computer science an extremely rewarding field and described their jobs as "the best in the world."

Despite the rewards, the aspirations of these women challenged them and their work environments, presenting them with a new equation to solve: how to juggle their roles as wives, mothers, and daughters while pursuing their professions as female technical researchers in a male-dominated workplace. They were often forced to compromise regarding institutional locations and ranks or to sacrifice promotions. While other women scientists sacrificed their jobs altogether, this particular group of women persisted along their career paths. One reason for their persistence may have been a shared love and excitement for their work and for learning. Over the years, these women developed defense mechanisms. They learned to play politics, to present themselves as professionals, and to be more assertive, and they relied on support networks in the form of local campus collaborations and professional societies, specifically the ACM.

The careers of these women computer scientists exhibited the same experiences reported in prior studies of scientific careers: (1) women had to solve the two-body problem of marriage and careers; (2) they benefited from affirmative action; (3) they were persistent; and (4) they learned how to react to gender discrimination (ignoring some incidents, de-emphasizing their own femininity, and avoiding tensions).[29] At the same time, they contributed their skills and talents to the scientific community, the classroom, and research laboratories. More importantly, they redefined the relationship between women and computing by demonstrating that women can have a rewarding technical career in computer science. Their personal accounts show "how individuals [are] able to carve out paths to success, locating the space for agency in lives constrained by gender roles."[30] Their careers and experiences testify to the power of women to overcome cultural stereotypes and leave a mark in computer science. Despite the tremendous obstacles, these extraordinary women have made history and changed the future for other women in computing.

29. G. Sonnert and G. Holton. 1995. *Who Succeeds in Science?* Rutgers University Press, New Brunswick, NJ.

30. J. Abbate. 2012. *Recoding Gender: Women's Changing Participation in Computing*, p. 152. MIT Press, Cambridge, MA.

5

The History and Purpose of Computing Curricula (1960s–2000s)

Sebastian Dziallas and Sally Fincher

No scene from prehistory is quite so vivid as that of the mortal struggles of great beasts in the tar pits. In the mind's eye, one sees dinosaurs, mammoths, and saber-toothed tigers struggling against the grip of the tar. The fiercer the struggle, the more entangling the tar, and no beast is so strong or so skillful but that he ultimately becomes ensnared. Development of model curricula that mesh computer science and engineering has, over the past decade, been such a tar pit, and many great and powerful beasts have thrashed violently in it.[1]

Producing curriculum reports is a small part of the overall work of the Association for Computing Machinery (ACM). But this work embodies issues that the organization faces as a wider body. In this chapter, we explore three areas through the lens of curriculum reports—the mobilization of volunteer effort, the negotiation with other professional bodies, and a necessary engagement in *defining the discipline* of computer science.

Founded in 1947, the ACM has been a major contributor to curricular standards in computing beginning in 1968 when the ACM was the first professional

1. M. C. Mulder. 1975. Model curricula for four-year computer science and engineering programs: Bridging the tar pit. *Computer*, 8(12): 28–33.

organization to release a model curriculum for computer science.[2] Since then, such curriculum guidelines have been released roughly every ten years. These have influenced educators at institutions across the U.S. at first by providing specific course sequences; in later years by specifying the material a computer science degree should cover and by offering examples of different course implementations.[3] Outside of the U.S., the guidelines have been referenced by the United Kingdom Quality Assurance Agency that specifies "what can be expected of a graduate in the subject, in terms of what they might know, do and understand at the end of their studies."[4] And the China Computer Federation (CCF) is currently developing a localized curriculum report for China, also based on the most recent 2013 ACM Computer Science curriculum guidelines.

The influence of the reports might lead an observer to assume that they are the product of a smooth, directed, and intentional process. However, the work leading to their creation has not always been without issue. In this chapter, we contend that the curriculum reports expose ongoing tensions that surface repeatedly in similar, although not entirely identical ways.

5.1 Always Volunteers: Coordinating Volunteer Efforts

To understand how curriculum reports shaped the computing discipline, we first need to understand the steps leading to their creation. Today, this is a well-established process: the ACM Education Board (or "Ed Board"), as the governing body overseeing the production of the curriculum reports, recruits a curriculum chair who then forms a steering committee with representatives from across the discipline. Each committee member then is responsible for a disciplinary knowledge area (such as "discrete structures" or "programming languages") and recruits community members for their specific content knowledge and experience. They form working groups which produce recommendations that are reviewed, edited, and assembled into drafts of the larger document.

2. W. F. Atchison, et al. 1968. Curriculum 68: Recommendations for academic programs in computer science: A report of the ACM Curriculum Committee on Computer Science. *Commun. ACM*, 11(3): 151–197.

3. S. Dziallas and S. Fincher. 2015. ACM Curriculum Reports: A pedagogic perspective. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, pp. 81–89. ACM, New York.

4. Quality Assurance Agency. 2016. Subject benchmark statement: Computing. Available at http://www.qaa.ac.uk/en/Publications/Documents/SBS-Computing-16.pdf.

Table 5.1 **Major changes between reports**

Society	Year	Curriculum Report	Major Changes
ACM	1968	Curriculum '68	First ACM report; focused on defining the subject and provided a suggested curriculum structure
IEEE-CS	1977	Curriculum in Computer Science and Engineering	First IEEE report; built on earlier NSF COSINE recommendations; provided a prescriptive model curriculum and reinforced aspects of hardware and logic design
ACM	1978	Curriculum '78	Significantly raised the profile of programming; introduced CS1-CS8 course sequence
IEEE-CS	1983	Model Program in Computer Science and Engineering	Updated the IEEE model curriculum; intended to be used in the evaluation of degree programs; included accreditation guidelines in an appendix
Joint	1984	Program Requirements and Accreditation	Not a curriculum report, but minimum program requirements for accreditation; first joint report between ACM and IEEE-CS; significantly increased expected course hours
Joint	1989	Computing as a Discipline	Aimed to distinguish computing from other disciplines; argued for a view beyond programming, including, e.g., design
Joint	1991	Computing Curricula 1991	Built on the *Computing as a Discipline* framework; introduced knowledge units and breadth-first curriculum
Joint	2001	Computing Curricula 2001	Reduced the size of the body of knowledge; returned to a more specific approach to course descriptions and included learning objectives
Joint	2008	Computer Science Curriculum 2008	Interim report; minor updates, including a section on security and "the computing crisis"
Joint	2013	Computer Science Curricula 2013	Advocated flexibility in relation to other disciplines; introduced curricular exemplars & division of core into tier 1 and 2; refined learning objectives by levels of mastery

The work of both the Education Board and the Curriculum Committee relies heavily on volunteer effort: the members of the Education Board and Curriculum Committee are all volunteers with their own individual goals and objectives. They are recruited by the committee chairs for a variety of reasons, perhaps to represent a specific disciplinary area, because of their work in computing education, or for their experience with previous curriculum efforts. Volunteers, of course, do not have to follow directives or "play by the book." And yet, they have a large role in the creation of these highly influential reports. Over time the ACM has repeatedly and in different ways worked to coordinate and to negotiate activities of its volunteers, from offering resources and funding for specific projects to the more direct influence of negotiating and approving committee appointments.

One example involves the first model curriculum of 1968. The thrust for the initial effort to incorporate computing into university curricula did not originate from within the ACM. Instead, it emerged in conversations among a group of directors of university "computer centers."[5] These computer centers were established in the late 1950s and early 1960s at universities in the U.S. to provide access to computing equipment and programming time—and to support teaching and research. And while industry supported the computer centers (typically giving discounts on equipment), their directors felt that formal curriculum recommendations should not be issued under the name of any particular company. At the time, the ACM had already established its Education Committee (a precursor to the Education Board), which compiled and distributed lists of institutions with computing-related degree programs and coordinated education-related efforts with local ACM chapters.[6]

In 1962, the group of computer center directors became the Curriculum Development Subcommittee (CDS) of the Education Committee to formalize work on a model curriculum. The importance of their effort to the ACM was clear.[7]

> This is your charter to pursue this task vigorously. I know that you will agree with me about the great importance of your undertaking. At this time this is perhaps one of the, if not "THE," most important activities ACM should engage in.[8]

But despite this enthusiasm, the CDS did not initially receive financial resources from the ACM and was forced to rely on external support for its meetings from IBM and the Southern Regional Education Board.[9] The committee also met in

5. W. C. Rheinboldt. December 8, 2014. Notes about the computer science curriculum 68. Letter to author (SD) January 21, 2016.

6. G. Heller. 1964. ACM education committee. Association for Computing Machinery Records, Charles Babbage Institute (CBI 205), Box 26, Folder 9B.

7. The chair of the Education Committee wrote to William Atchison, the newly appointed chair of the CDS. Atchison later recalled that he had been asked by Harry Huskey, ACM President from 1960–1962, "to look into computer science curriculum for the ACM" because of his involvement in early computer education efforts (W. F. Atchison. 1981. Computer education, past, present, and future. *SIGCSE Bull.*, 13(4): 2–6).

8. G. Heller. December 30, 1961. Letter to William F. Atchison. Appointment letter to the Curriculum Development Subcommittee. Charles Babbage Institute (CBI 205), Box 26, Folder 9A.

9. Rheinboldt, Notes about the computer science curriculum 68.

the vicinity of multiple ACM conferences and eventually secured funding from the National Science Foundation (NSF) to prepare the undergraduate curriculum recommendations.[10]

The ACM's interactions with volunteers dramatically expanded following the release of the first curriculum effort—known as *Curriculum '68*—when a wide range of efforts emerged under the Education Board. An activity report from 1973 lists many subcommittees, for community colleges, secondary schools, computer education in management, accreditation, and even a curriculum consulting service.[11] Even as responsibilities and projects increased, budgets did not, and volunteers reached the limits of their capacity to contribute. As the chair of the Education Board (1980–1984), David Kniefel, noted:

> Many educational activities could be revenue producers but those activities should not be at the expense of the important non-revenue activities such as curriculum development. Activities such as tutorials, program approvals and consulting, special publications, etc., can be at least self-supporting and perhaps even make a small profit to support other educational endeavors, but not on the strength of volunteers alone.[12]

To alleviate the strain on the volunteers, the Education Board passed a motion in 1972 expressing the need for a paid staff position focused on education at the ACM headquarters. "The continuing problem in the area of computer science education is appropriate staff support. Many education projects need more time and effort than is possible for volunteer personnel to give."[13] Nine years later, these issues still persisted when the Board issued a similarly worded call to fund a director of education. A report stressed: "Full-time support, publicity and coordination is *necessary*."[14]

In the late 1970s and early 1980s, the Education Board underwent organizational changes driven by ACM leadership. Some of these changes took a different approach to utilizing volunteers, suggesting that they were best deployed for specific, short-lived projects, as ACM President (1980–1982) Peter Denning noted: "This kind

10. E. J. Schweppe. 1990. On the genesis of Curriculum 68. In *ACM Annual Conference*, Washington, DC.

11. W. F. Atchison. 1973. Report by education board chairman to ACM Council. Charles Babbage Institute (CBI 205), Box 26, Folder 9B.

12. D. Kniefel. 1981. Annual Report of the ACM Education Board. Charles Babbage Institute (CBI 205), Box 26, Folder 9B.

13. Atchison, Report by education board chairman to ACM Council.

14. Kniefel, Annual report of the ACM Education Board, emphasis in original.

of work could replace our present approach to curricula: task forces are more limited in scope and duration, so that their participants are more likely to retain their enthusiasm until the job is done."[15]

By 1980, some of the volunteers on the Education Board had been involved in the creation of the curriculum reports for 15 years. The ACM leadership wanted to introduce new people and new perspectives to the Board as part of the organizational changes.

> My decision [to change the composition of the Education Board] arose partly from a desire to involve some new people in the Education Board and partly from my wish to move more strongly in the direction of secondary and community education.[16]

Of course, these issues with volunteer effort—of providing adequate financial support, retaining volunteer enthusiasm, and ensuring representation of diverse perspectives—do not eliminate its value. They expose the relationship between the volunteers and the ACM: the ACM played a tremendous role in the development of computer science as a discipline, but just as significant is the contribution of the volunteers working in the ACM's name.

5.2 Accreditation: Addressing a Threat to the Reputation of the Discipline

The ACM played an important role in establishing an accreditation mechanism for computer science programs at colleges and universities in the 1980s. The topic was another source of tension within the ACM: Should it be at all involved in the process of accreditation? Should it certify individual graduates or programs at institutions? Would the certification be based on its own criteria or that of another professional body? Others have explored these questions and the historical developments of accreditation in computer science more closely.[17] In this volume, for instance, Scott Campbell considers them in the context of professionalization in Canada. In

15. P. J. Denning. Letter to David Kniefel, June 16, 1980. Appointment letter to the Education Board. Charles Babbage Institute (CBI 205), Box 26, Folder 9A.

16. P. J. Denning. Letter to William Atchison, June 16, 1980. Charles Babbage Institute (CBI 205), Box 26, Folder 9A.

17. A. L. Russell. 2014. Accreditation and the boundaries of computer science, 1984–1999. Business History Conference, Frankfurt am Main, Germany. Abstract at http://www.thebhc.org/node/2921.

this chapter, we are particularly interested in the changing discussions at the ACM and in the way it positioned itself with regard to other professional or scientific societies.

Discussions on accreditation in the 1980s were driven by concerns about the quality of degree programs. The strong industry demand for graduates meant that institutions all over the U.S. had begun to offer computer science degrees. Not all of the programs provided adequate training, and many graduates were inadequately prepared.[18] The wide range of academic standards in computer science left both academics and industry professionals concerned about the reputation of the field. Accreditation—setting minimum criteria and requirements, approving degree programs, and certifying graduates—was widely seen as a way of addressing the problem. As Andrew Russell observed: "These professionals [working on accreditation] were motivated by the same types of fears that drove Curriculum 68, namely, the possibility that the reputation of their field as a whole would be discredited."[19]

The topic of accreditation surfaced repeatedly throughout the 1970s. The ACM and its Education Board coordinated efforts at multiple points: guidelines for "data processing schools" were initially released in 1969, followed by a set of minimum program criteria for use in regional accrediting activities in 1977.[20] But many of these efforts fell short because of a lack of funding or institutional support within the ACM. A committee report noted curtly: "The committee hopes to work on certification procedures for programmers as well as preparing guideline materials. The lack of available funds has curtailed these activities."[21]

The lack of funding was not coincidental: for most of the 1970s, the ACM maintained a policy of *not* actively engaging in the process of accreditation. For example, Walter Carlson, ACM president from 1970–1972, wrote:

> I suggest that the Accreditation Committee should not become involved with the "accreditation" or testing of the competence of individuals until very clear

18. M. C. Mulder and J. Dalphin. 1984. Computer science program requirements and accreditation. *Commun. ACM*, 27(4): 330–335.

19. Russell, Accreditation and the boundaries of computer science, 1984–1999.

20. ACM Accreditation Committee. 1977. Guidelines for a minimum program for colleges and universities offering 4 year bachelors degrees in computer science. Charles Babbage Institute (CBI 205), Box 24, Folder 30.

21. W. F. Atchison. 1973. Report by education board chairman to ACM Council. Charles Babbage Institute (CBI 205), Box 26, Folder 9B.

> evidence is provided to show that such testing is an essential, indispensable part of the process of accrediting educational or training activities.[22]

There were also concerns about the practicality of a large accreditation effort. While the goal of accreditation was to identify programs that represented sufficient academic standards in computer science, Peter Denning (ACM president 1980–1982) argued that institutions of all types and missions would have to participate for this kind of certification to be credible.

> Suppose the big institutions won't participate. How credible is the program if only small colleges and universities participate? How do we argue successfully that we have identified quality and excellence if the places most commonly thought of in that class won't participate? How do you make institutions feel they have something to gain by participating? The big ones probably won't feel that they have anything to gain, so a big sell job may be needed. (How is that to be accomplished?) The very weak places may like to be certified, but won't have a prayer. Some small number of marginal ones may be the only beneficiaries.[23]

These arguments about the need for an accreditation mechanism mirror the discussions taking place in Canada, as Scott Campbell observes in his chapter 8. As the problem of inadequately prepared graduates persisted in the United States,[24] other organizations worked to establish their own approaches to accreditation. When the ACM Education Board agreed to study the issue of accreditation again in early 1981,[25] it had to position itself with regard to these other efforts.

In the deliberations at the ACM, two contrasting approaches served as a model for accreditation in computer science. The American Chemical Society (ACS) licensed institutions to certify some—but not necessarily all—of their students upon meeting certain criteria. This approach stood in contrast to that taken by the Accreditation Board for Engineering and Technology (ABET) which accredited entire programs, rather than certifying individual graduates.[26]

22. W. Carlson. January 4, 1971. Letter to Charles Gulatto. Appointment letter to the Accreditation Committee. Charles Babbage Institute (CBI 205), Box 24, Folder 30.

23. P. J. Denning. August 5, 1981. Letter to John Dalphin. Charles Babbage Institute (CBI 205), Box 24, Folder 30.

24. W. Myers. 1986. Computing sciences accreditation pro and con. *Computer*, 19(6): 47–49.

25. E. Robertson. 1981. Education board meeting minutes. Charles Babbage Institute (CBI 205), Box 26, Folder 8.

26. ABET, founded in 1932 as the Engineers' Council for Professional Development, had already begun accrediting computer science and engineering degree programs in 1971. A later report

The newly appointed chair of the accreditation committee John Dalphin outlined detailed ideas for accreditation. In a significant departure from previous policy, he wrote: "It is proposed that the Association for Computing Machinery establish a program to accredit (approve) departments to allow certification of graduates from their programs in Computer Science and Information Systems."[27] The proposal aligned with the model of the ACS, in which individual departments would be responsible for the certification of their own graduates. Dalphin noted that this approach would provide "flexibility for departments in that not all students graduating will have to meet the minimum ideals established for certification."

As Thomas Misa writes in his introduction to this volume, the ACM and the IEEE Computer Society each played a significant role in defining computing as a discipline. While the ACM was exploring different approaches to accreditation, the Educational Activities Board of the IEEE Computer Society had formed a subcommittee which built on an earlier NSF-funded effort to develop a model curriculum in computer science. The IEEE curriculum was published in 1977 and was in many ways similar to that released by the ACM in the following year, but it placed a stronger focus on hardware and logic design and decreased the role of theory.[28] The IEEE Computer Society also provided guidelines for the evaluation of computer science and engineering programs to ABET.[29] And in 1983, the IEEE Computer Society released an updated version of its model curriculum.[30] As an indication of the growing importance of accreditation, the report included the ABET requirements for engineering programs as an appendix. In a second appendix, the report included a draft of "proposed program criteria for computer and similarly named engineering programs" that the IEEE-CS was preparing to submit to ABET.

By the time the IEEE Computer Society released its model curriculum in 1983, the educational context had developed. Previous reports from both societies were often seen as prescriptive documents that specified the exact material that should

by the Joint ACM/IEEE-CS Task Force noted that the accreditation process ABET employed was "only appropriate, however, for computer science programs evaluated as engineering programs" (M. C. Mulder and J. Dalphin. 1984. Computer science program requirements and accreditation. *Commun. ACM*, 27(4): 330–335).

27. J. Dalphin. July 21, 1981. Letter to David Kniefel. Charles Babbage Institute (CBI 205), Box 24, Folder 30.

28. IEEE Computer Society Model Curriculum Subcommittee. 1977. A curriculum in computer science and engineering: Committee report.

29. Mulder and Dalphin, Computer science program requirements and accreditation.

30. IEEE Computer Society Model Program Committee. 1983. The 1983 IEEE computer society model program in computer science and engineering.

be taught and in which order.[31] The IEEE-CS 1983 model curriculum guidelines were aimed at the evaluation of degree programs.

> The faculty can use this document [the model curriculum] as a reference to evaluate the strengths of their own program, as a source of opinions of their colleagues in industry, government, and academia, and as a guideline for updating their own curricula.[32]

The changing circumstances surrounding model curricula, the deliberations inside the ACM, and the work of the IEEE Computer Society provided common ground to jointly explore issues surrounding the accreditation of computer science degree programs. Both societies agreed to support an accreditation mechanism for computer science programs. In the spring of 1982 they formed a joint task force with a wide ranging charter to establish minimum criteria for computer science programs and to recommend the structure of an accreditation mechanism.[33]

When the joint task force released minimum program requirements and criteria in 1984, they specified the creation of a new Computer Science Accreditation Commission (CSAC).[34] It was designed to parallel the structure of ABET, which organizes its work around a number of disciplinary "commissions," such as the Engineering Accreditation Commission. However, one of the remaining open questions was the umbrella CSAC would operate under. ABET, with its role in accrediting engineering programs, was seen by some academics in computer science, and in particular those at liberal arts institutions, as an inappropriate choice. As Michael Mulder, a member of the Joint ACM/IEEE-CS Task Force, observed: "The [IEEE] Computer Society recommends that CSAC operate under the ABET organization; the ACM position is not clear at this time, in that concern over excessive engineering influence exists."[35]

31. S. Dziallas and S. Fincher. 2015. ACM Curriculum Reports: A pedagogic perspective. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, pp. 81–89. ACM, New York.

32. IEEE Computer Society Model Program Committee. 1983.

33. M. C. Mulder. March 14, 1983. Letter to Edward Ernst. Charles Babbage Institute (CBI 205), Box 24, Folder 30; D. Kniefel. April 8, 1983. Letter to ACM Extended Executive Committee. Charles Babbage Institute (CBI 205), Box 24, Folder 30.

34. M. C. Mulder and J. Dalphin. 1984. Computer science program requirements and accreditation. *Commun. ACM*, 27(4): 330–335.

35. Mulder, Letter to Edward Ernst.

ABET appeared to be eager to work with the joint task force and to incorporate computer science accreditation into its existing processes.[36] Despite a willingness to address concerns, the idea of incorporating the CSAC into ABET ultimately did not succeed. Tom Cain, also involved in the Joint Task Force, later recalled:

> I joke that some members of ABET weren't sure that electrical engineering was real, let alone computer engineering, let alone computer science. So the only choice was to try and develop a separate organization, and we put together what became known as the Computing Sciences Accreditation Board or CSAB.[37]

CSAB was set up to mirror ABET's structure, although it only had one commission, the CSAC. And although it was not part of ABET, it received help in the form of office space and the employment of its executive director.[38] In 1998, ABET and CSAB signed a memorandum of understanding with the intention of eventually integrating CSAB into ABET; this was accomplished in 2001.

Accreditation exposed tensions within the ACM. The ACM moved from initial reluctance to support accreditation towards using it as a mechanism to address a threat to the reputation of the discipline. Its views on the influence of other disciplinary concerns also shifted—and with it, the positioning with regard to other societies. The question of whether computer science was an engineering discipline, a science, or a field of its own occupied many task forces and committees.

5.3 Definition of a Discipline

Throughout the past five decades, computing has engaged repeatedly in discussions about its definition as a discipline and where its boundaries lie. It has positioned itself externally against other disciplines, such as mathematics and engineering, and also internally, against a spectrum of practice ranging from liberal arts at one end and computer engineering at the other.

These discussions can be seen as far back as the work on the first model curriculum in the 1960s. As computer science was, especially at that time, an emerging

36. E. Ernst. March 30, 1983. Letter to Michael C. Mulder. Charles Babbage Institute (CBI 205), Box 24, Folder 30.

37. J. T. Cain. 2009. Interview by Sheldon Hochheiser. IEEE History Center.

38. G. Engel, J. Impagliazzo, and P. LaMalva. 2009. A brief history of the Computing Sciences Accreditation Board (CSAB): Promoting quality education in the computing fields. In *2009 IEEE Conference on the History of Technical Societies*, pp. 1–8.

discipline, many of the members of the curriculum committee were numerical analysts who had a strong background in mathematics. As a result, even among the group that ultimately produced the *Curriculum '68* report, the need for curriculum guidelines in computer science at all had first to be established.

> There was not even complete agreement that undergraduate programs were desirable, in fact, one of the recommendations of the Mathematical Content and Prerequisites Subcommittee was that we should not make recommendations for such programs for fear of encouraging them. There did develop, however, a general consensus that such programs were inevitable. On the basis of this agreement, it was decided that strong guidelines were needed in order to improve the quality of the programs that would evolve.[39]

Building on this agreement to develop "strong guidelines," the first curriculum reports were highly prescriptive and aimed to set and to defend disciplinary boundaries while providing guidance to newly established departments. We previously characterized this prescriptive role as *curriculum as a weapon*.[40] despite the initial agreement, the discussions continued even beyond the release of preliminary recommendations.

> The *classification problem* occupied a great deal of time in the general meetings of the committee. There were long and serious debates as to just what constituted the field of computer science and how it should be broken down into subareas. . . . The committee almost got bogged down on this matter and was having difficulty in proceeding with the development of the curriculum.[41]

To address this problem, two members of the committee developed what they called the "ultimate compromise" consisting of three parts: the committee agreed to view computer science as its own field; it decided to focus the mathematical requirements in the model curriculum on discrete mathematics, rather than calculus and analysis as some committee members were suggesting; and it did not go into depth on the specific applications of computer science in engineering and business.[42] The resulting report, which was published in 1968 soon after the com-

39. E. J. Schweppe. 1990. On the genesis of Curriculum 68. In *ACM Annual Conference*, Washington, DC.

40. S. Dziallas and S. Fincher. 2015. ACM curriculum reports: A pedagogic perspective. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, pp. 81–89. ACM, New York.

41. Schweppe, On the genesis of Curriculum 68, emphasis in original.

42. W. C. Rheinboldt. January 21, 2016. Letter to Sebastian Dziallas.

mittee reached its compromise, split computer science into three major subject areas ("information structures and processes," "information processing systems," and "methodologies") and identified two related disciplines ("mathematical sciences" and "physical and engineering sciences").[43] The report focused on what the authors viewed as the fundamentals of the discipline and what was necessary to prepare students for graduate study in computer science.[44]

The importance of, and influence of, other disciplines is reflected in the different approaches of the two major societies: ACM and IEEE Computer Society. In the 1970s and 1980s, the discipline had to define its relationship to engineering; bridging the gap between "computer science" and "computer engineering" initially proved difficult. The ACM and IEEE-CS approached each other carefully: the ACM went so far as to establish an "IEEE Liaison" held by Gerald Engel.[45]

ACM and IEEE Computer Society eventually came together in 1989 to publish the *Computing as a Discipline* report (the IEEE sent one representative, Michael Mulder, who had already been involved in the Joint Task Force on accreditation). The report explicitly aimed to address questions surrounding the definition and role of computer science.[46] Unlike the early, prescriptive curriculum reports, it presented a framework "to answer the nagging education questions of the day, is computer science engineering? Science? Mathematics? Where does it fit in a university?" As Peter Denning, the chair of the report, recalled: "I just did not want us to become the victim of other people's stories about us. There was so much we could do for ourselves. I wanted to help computing find its own voice."[47]

The joint committee explicitly included computer science and computer engineering in its view of the discipline and used the term *computing* to refer to both fields together.[48] It also showed concern for different perspectives: in developing paradigms of the discipline, it introduced a framework consisting of three concepts—theory, abstraction, and design. Of these three paradigms, the

43. W. F. Atchison, et al., 1968. Curriculum 68: Recommendations for academic programs in computer science: A report of the ACM Curriculum Committee on Computer Science." *Commun. ACM* 11(3): 151–197.

44. Schweppe, On the genesis of Curriculum 68.

45. W. F. Atchison and R. H. Austing. 1979. Annual report of the Education Board. Charles Babbage Institute (CBI 205), Box 26, Folder 9B.

46. D. E. Comer, D. Gries, M. C. Mulder, A. Tucker, A. J. Turner, and P. R. Young. 1989. Computing as a discipline. *Commun. ACM*, 32(1): 9–23.

47. P. J. Denning. 2015. Interview by Sebastian Dziallas.

48. Denning, et al. Computing as a discipline.

authors attributed theory to mathematics, abstraction to the "experimental scientific method," and design to engineering. They wrote:

> Many debates about the relative merits of mathematics, science, and engineering are implicitly based on an assumption that one of the three processes (theory, abstraction, or design) is the most fundamental. Closer examination, however, reveals that in computing the three processes are so intricately intertwined that it is irrational to say that any one is fundamental.[49]

Peter Denning later recalled the emergence of the term "design" in discussions among committee members: "The term design was actually a new insight. I remember a meeting in which Mike Mulder was shaking his head on account of all the attention we had been paying to theory and abstraction."[50] These were of course the concepts more closely aligned with the perspective of the ACM.

> He said that emphasis will never fly with his engineering colleagues. Then he said, "Design would work. It is a deep value of engineering and appears in the accreditation criteria. It is integral to software engineering." That moment of insight caused the entire committee to coalesce on design rather than engineering, and we sketched out a design paradigm. . . . Mike Mulder reported later that his IEEE colleagues accepted this formulation and from that moment we had a solid basis of working together.[51]

The *Computing as a Discipline* report formed the foundation of the subsequent 1991 ACM/IEEE-CS curriculum report.[52] This effort brought people from both societies together once again. However, this time there were particular difficulties in bridging differences between engineering and liberal arts programs: differences in participants' backgrounds lead to differences in perspectives, which contributed to tensions within the group. Opinions varied widely on when to introduce concepts such as P/NP, for example, or whether physics should be a compulsory course for computing students.

> The notion that one might have a curriculum that was more flexible and had lower requirements than is typical in an engineering school, some of them found that difficult to accept and thought that it just meant you were watering things down, that it wasn't a real curriculum, and so there were a number of strains.

49. Ibid.

50. P. J. Denning. 2015. Interview by Sebastian Dziallas.

51. Ibid.

52. A. B. Tucker, editor. 1991. Computing curricula 1991. *Commun. ACM*, 34(6): 68–84.

> The ACM and the IEEE people tended to have different points of view. Obviously, there was a range in there, but there was often a fair amount of tension.[53]

The liberal arts agenda was particularly prominent within the ACM community, and persistent representations were made to get the liberal arts perspective embodied in the curriculum committees. Its influence dates back to the release of the 1977 accreditation guidelines, when Education Board meeting minutes reveal that the chairman of the accreditation committee was especially asked to "look into the feelings of small schools that the accreditation guidelines were too hard."[54]

In a letter to *Communications of the ACM*, Allen Tucker and Norman Gibbs highlighted emerging problems for departments at liberal arts institutions offering B.A. degrees in computer science.[55] They feared that these programs would have to conform to B.S. requirements. In response to the letter, other academics added their voices:

> For years a view has been expressed by a small group of people, who are primarily located in computer science departments within schools of engineering within large universities, that computer science is confined to those programs and departments that emphasize research and scholarly publication. Programs that "merely" teach the theory and its application are not professional and shouldn't be allowed to call themselves "computer science."[56]

One of the biggest concerns for the representatives of small and liberal arts institutions was simply the size of the curriculum and how they would be able to cover a defined computer science curriculum: many had only a limited number of available course hours and few instructors.[57] The 1984 accreditation guidelines marked a noticeable increase in required course hours from the previously released Curriculum '78 report.

> At UC Berkeley, the computer science major for liberal arts students . . . devotes 33 hours to computer science itself, far less than the 45 hours specified by the

53. K. B. Bruce. 2015. Interview by Sebastian Dziallas.

54. ACM Education Board. 1978. Education Board meeting minutes. Charles Babbage Institute (CBI 205), Box 26, Folder 8.

55. A. B. Tucker and N. E. Gibbs. 1984. On accreditation. *Commun. ACM*, 27(5): 411–412.

56. T. R. Harbron. 1984. More on accreditation. *Commun. ACM*, 27(10): 988.

57. S. Dziallas and S. Fincher. 2015. ACM curriculum reports: A pedagogic perspective. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, pp. 81–89. ACM, New York.

> CSAB criteria. In comparison, the ACM model curriculum requires only 33 to 36 hours in this area. . . . [58]

Liberal arts colleges, then, were unsatisfied with the status quo of curricular guidelines. As Henry Walker and Charles Kelemen observed, the problem was that "the same recommendations were to apply to technical schools, research-oriented universities, and liberal arts colleges."[59] These concerns led to the emergence of the Liberal Arts Computer Science Consortium (LACS), an alliance of concerned individuals from liberal arts institutions. In 1986, with support from the Sloan Foundation, they published the first "Model Curriculum for a Liberal Arts Degree in Computer Science."[60] It provided suggestions for how an institution with a small computer science faculty would be able to offer a B.A. degree. The curriculum was highly prescriptive, even including a detailed description of a teaching load distribution for departments with as few as three faculty members.

The group aimed to provide other liberal arts institutions with the resources to establish computer science programs. Among the initial questions considered by the members of LACS were: "What kind of curriculum would be appropriate and realistic in the small liberal arts college environment? How could we attract faculty to this kind of environment?"[61]

Following the release of the 1991 curriculum guidelines, LACS again felt the need to develop and to release its own recommendations. And while the 2001 task force explicitly worked to reduce the size of the mandatory part of the curriculum,[62] the 2001 ACM curriculum was symmetrically followed by the release of LACS recommendations in 2007.[63]

> In the 2001 there was an effort in the task force to be broader and think of more perspectives. But ultimately LACS concluded it was a nice effort, but it

58. W. Myers. 1986. Computing sciences accreditation pro and con. *Computer*, 19(6): 47–49.

59. H. M. Walker and C. Kelemen. 2010. Computer science and the liberal arts: A philosophical examination. *ACM Trans. Comput. Educ.*, 10(1): 2:1–2:10.

60. N. E. Gibbs and A. B. Tucker. 1986. A model curriculum for a liberal arts degree in computer science. *Commun. ACM*, 29(3): 202–210.

61. K. B. Bruce, R. D. Cupper, and R. L. S. Drysdale. 2010. A history of the liberal arts computer science consortium and its model curricula. *ACM Transactions on Computing Education*, 10(1): 3:1–3:12.

62. Joint Task Force on Computing Curricula, editor. 2001. Computing Curricula 2001. *ACM Journal of Educational Resources in Computing*, 1(3es).

63. For an overview, see Bruce, et al. A history of the liberal arts computer science consortium and its model curricula.

> really didn't get the job done in terms of what would make sense in a liberal arts perspective.[64]

In 2013, the computing curriculum Steering Committee addressed the issue of the size of the curriculum early on.

> Something we were very cognizant of from the beginning is how do we create these guidelines that contain new material, but can't require more hours of instruction? That is what creates some of the real challenge: if you're putting new stuff in, what's the old stuff that comes out? You're always going to upset someone when you take old stuff out, because if it's their stuff, they're going to be upset. But luckily, we found a structure, with this tiered structure, that worked.[65]

The 2013 curriculum report introduced a two-tiered structure.[66] While previous reports had distinguished between core and elective materials in the body of knowledge, the 2013 report further separated the core into tier 1 and 2. Material in tier 1 is seen as fundamental to any degree program in computing, and thus essential. At the same time, the 2013 report acknowledges that not every degree program must necessarily include all the content in tier 2. The liberal arts response to the 2013 report has been notably different.

> For the 2013, with two of the three curricula exemplars for four-year programmes coming from Liberal Arts, we're really pretty pleased that our perspectives are represented in a meaningful way. I don't believe there's expectation there will be a follow-up consortial [LACS] response, because effectively then that's been incorporated already into what's there.[67]

Despite this positive response to the 2013 curriculum report, representatives from liberal arts institutions have subsequently formed a Committee on Computing Education in Liberal Arts Colleges under the auspices of the Computing Education Special Interest Group of the ACM (SIGCSE). While LACS has represented liberal arts interests to date, its membership has been limited to a small number of institutions. Going forward, this new committee intends to continue to influence matters of computing education: by providing a more representative perspective

64. H. M. Walker. 2015. Interview by Sebastian Dziallas.

65. M. Sahami. 2015. Interview by Sebastian Dziallas.

66. Joint Task Force on Computing Curricula. 2013. *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. ACM, New York.

67. Walker, Interview by Sebastian Dziallas.

and by focusing on "the development of a national "voice" for computing education within liberal arts colleges."[68]

5.4 Conclusion

In this chapter, we considered three areas in the context of curriculum reports—the role volunteers play in the creation of the reports and how the ACM coordinates their work; how the ACM's stance on accreditation developed over the past five decades; and discussions on the boundaries of the discipline, on what does and does not constitute computer science. Each of these areas has wider implications beyond the curriculum efforts themselves.

First, especially in regard to education, the ACM relies on volunteer effort. This reliance means that effort must not be abused (volunteers should be recruited for short term, and not over-used, to retain enthusiasm) but also that it cannot be compelled. If individuals have a personal project to propose, or specific agenda to pursue, it is relatively easy for them to do so. And achieving aims is a large part of the reward for volunteering.

> I think, in fact, these kinds of things [the curriculum reports] are 10% respect and honor, and 80% to 90% hard work that nobody really wants to do.[69]

Second, the process of establishing an accreditation mechanism reveals the importance of interacting with other organizations and societies. Third, there are still ongoing debates about the scope and definition of the discipline, what it consists of, and what should be taught. However, as computing has matured and its technologies have become pervasive, the site of these arguments has moved away from boundary disputes with other disciplines (mathematics and engineering). As programming is now an important practice to many other subject areas, curricular efforts are adopting a "big tent" view in which computing contributes to other subject knowledge to form new disciplinary areas in their own right (such as computational biology). The arguments now focus on what parts of computing are important when combined with other disciplines. The role of programming has also expanded beyond the boundaries of computing to the desiderata that all children should learn to code as part of their education—or at the least be exposed to "computational thinking."[70] Indeed, in his 2016 State of the Union address, United

68. D. Baldwin, A. Brady, A. Lawrence, and H. Walker. 2015. SIGCSE Committee on Computing Education in liberal arts colleges. Available at http://sigcse.org/sigcse/programs/committees/liberal.

69. S. Simonson. 2015. Interview by Sebastian Dziallas.

70. J. M. Wing. 2006. Computational thinking. *Commun. ACM*, 49(3): 33–35.

States President Barack Obama called for every student to learn computer science in school. This expansion of computing—and especially programming—to other subject areas and to secondary education has brought concern that the importance of programming now overshadows other parts of the discipline. Referring to the 1991 curriculum report, Peter Denning noted:

> Do we want to emphasize programming and get the students deeply into the practice of programming? Or do we want to emphasize the science and engineering and view programming as one of several important computing practices? I think that debate is still going on today.[71]

Some of these tensions, then, are reminiscent of earlier discussions within the discipline, while others are in uncharted territory. And as in the early days computing, they will continue to require both liaison with other bodies and the creation of new compromises.

5.5 Acknowledgments

This work was supported through the award of a 2015 ACM History Fellowship. A version of this chapter was presented at the annual meeting of the Society for the History of Technology in Albuquerque, NM in 2015. And parts of this chapter are based on a paper entitled “ACM Curriculum Reports: A Pedagogic Perspective” published at the 2015 International Computing Education Research conference.[72]

We are grateful to the 14 participants from different curriculum efforts who participated in interviews with us for this project. We would like to thank the Charles Babbage Institute at the University of Minnesota, the Stanford University Archive, and the University of Maryland Archives for helping us access ACM archive materials and for permitting us to include parts in this work, as well as the Computing Educators Oral History Project and the IEEE History Center for allowing us to include excerpts from their oral history interviews.

71. P. J. Denning. 2015. Interview by Sebastian Dziallas.

72. S. Dziallas and S. Fincher. 2015. ACM curriculum reports: A pedagogic perspective. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, pp. 81–89. ACM, New York.

6 “Deeply Political and Social Issues”: Debates within ACM 1965–1985

Janet Toland

6.1 Introduction

The social conservatism that followed the end of World War II was partially dissolved by a new spirit of optimism in the 1960s.[1] In certain sections of society this led to a growing skepticism towards authority that was typified by the rise of the so-called counter culture. In the U.S. one of the most polarizing issues was the ongoing war in Vietnam which led to a growing political rift between “hawks” and “doves.” The scientific community was not exempt from these influences. During the 1960s many scientists began to reassess the relationship between their professional lives and their personal beliefs, and although some felt that there should be a clear separation between the two, a prominent and outspoken faction felt it was their duty to strive to influence policy on social issues related to their area of professional expertise. Computing was subject to the same pressures as the traditional science fields like physics and chemistry, and, as a newer discipline without the benefits of precedent, arguably felt them even more keenly.

This chapter explores the tensions within different parts of ACM as it reacted to the political and social events of these turbulent decades. It looks at how computer professionals viewed themselves and their responsibilities to the wider world, both as individuals and as members of a professional society. Much of the debate concerns *which* political and social issues were appropriate for a group such as ACM to comment on, and how that argument was negotiated between different parts of

1. R. Osborne. 2011. *Civilization: A New History of the Western World*. Random House, New York.

Figure 6.1 **ACM president Jean Sammet presents Outstanding Contribution Award to W. Smith Dorsey (June 1976).** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

ACM, notably the Executive Committee, the special interest groups, and the Public Policy Committee. As ACM president Jean Sammet put it: "Those at one end wish ACM to have no involvement with anything other than purely scientific, educational and technical matters, while those at the other end wish ACM to have a strong involvement with all aspects of life."[2]

The focus of this chapter is on ACM as a group of computer professionals who wanted to educate the public and influence policy through the existing democratic system. Rebecca Slayton in *Arguments that Count* argues that computing is not a profession as such but a looser group of individuals who share expertise in a partic-

2. J. Sammet. 1976. ACM President's letter: Relation of ACM to the world outside ACM. *Commun. ACM*, 19(2): 57.

ular discipline.[3] Generally, a discipline is regarded as a coherent and accepted body of knowledge, while a profession has an intellectual core plus some institutional means of maintaining standards and policing membership. Although computing has never had the same level of control that professions such as medicine and law have, it was clear that those computer scientists who chose to join ACM considered themselves to be part of a fledgling profession with strong ethical standards and were willing to volunteer their time to ensure that the relatively new discipline of computing became well respected, both within the scientific community and by the general public. As long-term ACM administrator Irene Hollister put it: "Because early on they were strictly all professional people. They were so intelligent as a group, and also their standards were very high, their intellectual standards and their moral standards."[4] Even though members had very different opinions about what were—and were not—suitable topics for computer scientists to involve themselves with, they were all in agreement that any attempt to influence the public or government should be through legitimate means that were a well-established part of the democratic process. Although individuals may have shared some of the political opinions of the counter culture, they were clear that they wanted to work within the rules of the existing establishment and influence policy from inside the system.

The records of the Special Interest Group on Computers and Society (SIGCAS), active for over 40 years, provides evidence for evaluating professionalism and social activism in the ACM. The main focus of this chapter is on the development of SIGCAS as a special interest group and its changing emphasis of interest. Sources of evidence include records of SIGCAS-sponsored panels and technical sessions at the national ACM conferences and material from the SIGCAS newsletter *Computers and Society*. I also draw on other ACM materials such as the papers of Edmund Berkeley, a co-founder of ACM renowned for his social activism, and Daniel McCracken who chaired the Committee on Computers and Public Policy from 1973–1976. Oral history interviews with former ACM presidents Peter Denning,[5] Daniel McCracken,[6]

3. R. Slayton. 2013. *Arguments That Count: Physics, Computing, and Missile Defense, 1949–2012*, p. 4. MIT Press, Cambridge, MA.

4. I. Hollister. 2007. Oral History Interview No. 10, by Jennifer Light, p. 17. ACM Digital Library, New York.

5. P. J. Denning. 2007. Oral History Interview No. 20, by Arthur L. Norberg. ACM Digital Library, New York.

6. D. D. McCracken. 2008. Oral History Interview No. 11, by Arthur L. Norberg. ACM Digital Library, New York.

and Anthony Ralston[7] and ACM administrator Irene Hollister[8] illustrate general ACM activity, as does an interview with Computer Professionals for Social Responsibility (CPSR) founders Severo Ornstein and Laura Gould.[9]

This chapter will set the scene by looking at three key issues that caused controversy within ACM: the use of computer technology in warfare; the Equal Rights Amendment; and the human rights of Soviet computer scientists. It then looks more specifically at the development of two inter-related groups within ACM set up to deal with social and political issues, the Committee on Computers and Public Policy (CCPP) and the Special Interest Group on Computers and Society (SIGCAS).

6.2 1969: A Question of Importance

From its formation in 1947, a number of significant individuals within ACM have wanted the association to take a strong stand on "deeply political and social issues." Edmund Berkeley had a long history of protesting against the use of nuclear technology in warfare and the role that computer technology played in the development of nuclear weapons. As noted in Nelsen's chapter 10 in this book, he was also a strong advocate for social responsibility in the computing profession, and used his journal *Computers and Automation* as a platform to debate political and social issues through various articles and editorials,[10] often written by himself or using his pseudonym of Neil Macdonald.[11] Daniel McCracken was another ACM member concerned about using computer technology in weaponry; in 1965 he formed Computer Professionals against Anti-Ballistic Missiles (ABM)[12] as an ad hoc organization of around 500 people, with an executive committee of four including himself, Paul Armer, Joseph Weizenbaum, and Greg Williams. The group was formed to op-

7. A. Ralston. 2006. Oral History Interview No. 6, by William Aspray and Bernard de Neumann. ACM Digital Library, New York.

8. Hollister, Oral History Interview No. 10.

9. S. Ornstein and L. Gould. November 17, 1994. Oral History Interview, by Bruce Bruemmer. Charles Babbage Institute (OH 258). Available at http://purl.umn.edu/107336.

10. See Edmund C. Berkeley papers, Charles Babbage Institute (CBI 50), Box 10, Folder 72. Social responsibilities of computer people. *Computers and Automation*, January 11, 1962.

11. B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals*, p. 143. Association for Computing Machinery and Morgan & Claypool, New York.

12. Slayton, *Arguments That Count*, 119.

pose the Safeguard antiballistic missile system,[13] as McCracken and his colleagues strongly believed that the computing technology at the heart of the Safeguard system would not be capable of performing the complex job assigned to it. They argued that their opposition was on purely technical grounds. “Almost all of us were flaming liberals too, but in the minds of our opponents, we would not have been in favor of it if we thought it would work. We never denied that, but we said, Look that’s not what we’re talking about here. Even if you think it’s a good idea, don’t do it. It’s going to be a waste of money. It’s not going to work.”[14] McCracken used the royalties from publishing a number of bestselling books on computing to travel up and down the country giving speeches to various organizations on behalf of Computer Professionals against the ABM: “that was a lot of fun. It was a lot of work.”

Berkeley was formally recognized as the key founder of ACM on its 25th anniversary, but he was regarded as something of a maverick and in later years was kept very much on the sidelines. In contrast, McCracken held a number of senior positions within ACM, including chair of the Public Policy Committee from 1973–1976 and the presidency from 1978–1980, and he remained highly influential. Even though other key figures of the period such as Jean Sammet and Peter Denning, ACM presidents respectively in 1974–1976 and 1980–1982, often disagreed with his opinions, they respected his political skills. As ACM member Bruce Van Atta commented when writing to McCracken in the aftermath of the Turchin issue (to be discussed later): “Just a very brief note of congratulations on obtaining an ACM resolution concerning Dr. Valentin Turchin. Although opposed to it as you know, I cannot help but marvel at the adept manner in which you handle sensitive issues.”[15]

By 1969 opinions over the Vietnam War were coming to a head. Several ACM members called for the association to issue a statement against the war, while other members strongly opposed this, feeling that expressing such opinions was well outside the constitution of ACM. There had also been dissent over whether the ACM national conference planned for 1971 should be located in Chicago; a number of ACM members felt it should be moved to another location in response

13. The Safeguard anti-ballistic missile system was designed to protect the U.S. Air Force’s Minuteman land-based intercontinental ballistic missiles from attack. Safeguard was developed from the Nike Zeus system designed by Bell Labs.

14. D. D. McCracken. Oral History Interview No. 11, p. 25.

15. D. D. McCracken Papers, Charles Babbage Institute (CBI 43), Box 5, Folder 7: Turchin Affair. Letter from Bruce W. Van Atta to Dan McCracken. November 14, 1975.

to the riots and allegations of police misconduct[16] that occurred during the 1968 Democratic National Convention.[17] These issues raised the question of whether the ACM's constitution should be revised to permit comment on what were termed "deeply political or social issues."[18] The question: "Shall the Constitution of the ACM be revised to permit the Association to comment or action on deeply political and social questions?" was posed to members in a framework which involved the possibility of ACM taking a position on the Vietnam War and considering whether the ACM annual conference in 1971 should be moved from Chicago. This "Question of Importance" was referred to the ACM membership for a general vote in April 1969 and was resoundingly defeated: the 'No' vote was 7,938 as compared to 2,059 for "Yes". This vote was a "watershed" moment for ACM, and is also discussed in detail in Nelsen's chapter. The ACM president at the time, Bernard Galler, expressed amazement at the large number of letters he had received protesting that such a vote had even been taken. In a letter to members informing them of the outcome, Galler encouraged members to get involved with politics personally, rather than through their professional association.[19]

6.3 1972: The Equal Rights Amendment

Peter Denning, ACM president from 1980–1982, was always clear that he wanted ACM to be apolitical: "A sort of Switzerland among professional societies."[20] He noted that ACM had more than its fair share of political activists who all wanted ACM to become a spokesperson for their political agendas, but Denning felt it was potentially destructive for a young society to get involved in political activism. His vision of ACM was of an organization where people of all political stripes could

16. The 1968 Democratic convention followed a year in which there were riots in over 100 cities following the assassinations of Martin Luther King and Robert Kennedy. 10,000 demonstrators from groups such as the National Mobilization Committee to End the War in Vietnam, the Youth International Party (Yippies), and Students for a Democratic Society gathered in Chicago for the convention where they were met by 23,000 police and National Guardsmen. On August 28, 1968 a so-called riot took place which ended in front of the Hilton Hotel.

17. J. Sammet. 1975. ACM President's letter: Guidelines for deeply political/social issues. *Commun. ACM*, 18(5).

18. R. Slayton. 2013. *Arguments That Count: Physics, Computing, and Missile Defense, 1949–2012*, p. 120. MIT Press, Cambridge, MA.

19. B. Galler. 1969. ACM President's letter: I protest. *Commun. ACM*, 12(8).

20. P. J. Denning. 2007. Oral History Interview No. 20, by Arthur L. Norberg, p. 95. ACM Digital Library, New York.

Figure 6.2 **Peter Denning and Audience at ACM Annual Meeting (Houston 1993).** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

convene and do technical and educational work together. He felt it was especially risky for the Executive Committee to take a public position on political issues since they could not be sure that their views were shared by all members.

During an interview historian Arthur Norberg pressed him on these points and his own political views.[21] Denning reiterated that his main arguments for keeping ACM away from deeply political and social questions were that ACM was still immature compared to other professional societies and that its credibility depended on its staying focused on developing technical expertise. He also acknowledged

21. Ibid.

his discomfort about trying to represent the views of members who hadn't been polled. He also explained the ramifications of ACM's tax status as a 501(c) (3) organization which meant it was supposed to be primarily engaged in matters for the public good; lobbying to influence legislation is explicitly forbidden. In her 1976 President's letter, Jean Sammet explains that according to IRS's classification ACM is actually a scientific and educational society rather than a professional society.[22] As Sammet made clear, the difference is that a scientific and educational society has its main concern for the public good and education and science, whereas a professional society is operated primarily for the benefit of a particular profession or industry.

Denning did point out that if he strongly believed that a public policy issue had weaknesses he would speak out against it. For example, he had joined forces with Daniel McCracken and Jean Sammet to develop position papers on issues such as the use of social security numbers as unique identifiers in computer systems, and whether software engineering could produce a workable Strategic Defense Initiative, popularly known as the "Star Wars" system. However, he was clear that he wanted to keep these as position papers rather than political recommendations.[23]

Denning's most serious disagreement with McCracken concerned the Equal Rights Amendment bill which was before the U.S. Congress in 1972.[24] McCracken wanted ACM to take a position squarely favoring the amendment. Denning thought it was outside the scope of ACM. There was a strong contingent of social activists who wanted ACM to take a stand and were even willing to risk ACM's tax exempt status. The situation got quite nasty. Denning was called a "chauvinistic pig" and informed he was using his official position to block ACM from doing the right thing. Jean Sammet also got "roasted." She personally supported the amendment but she did not think it was appropriate for ACM to get involved in what she regarded as lobbying. In taking this stand she was out of line with other female ACM volunteers, who wanted the organization to support this issue.[25]

22. J. Sammet. 1976. ACM President's letter: Relation of ACM to the world outside ACM. *Commun. ACM*, 19(2).

23. Denning, Oral History Interview No. 20, p. 96.

24. The Equal Rights Amendment bill (ERA) was a proposed amendment to the United States Constitution to ensure equal rights for women. Originally proposed in 1923, it passed both houses of Congress in 1972 and was submitted to the state legislatures for ratification. The required number of ratifications was not achieved and the amendment was never implemented.

25. Denning, Oral History Interview No. 20, p. 98.

6.4 1975: The Turchin Issue

Valentin Turchin was a Russian computer scientist who for some years had increasingly serious problems with Soviet authorities because of his political activities. Turchin was chairman of the Moscow chapter of Amnesty International and had made statements in support of Soviet dissident Andrei Sakharov.[26] Turchin was formerly head of a laboratory at the Institute for Automated Systems in the Building Industry, but had been dismissed in July 1973 and since then no-one had been allowed to hire him. His apartment had been searched and he was subjected to six lengthy interrogations; he felt he had no alternative but to leave the USSR. Turchin had been offered a post as a visiting scholar in the mathematics department at Columbia University, but the Soviet government would not give him the necessary exit visa. Turchin was a member of the ACM, as an American friend had recently paid his dues, and the ACM Executive Council was asked to write a statement to the Soviet authorities in support of him being granted permission to enter the U.S.

The debate that this brought about at the ACM Council meeting was widely reported on[27] and discussions were polarized. Gerry Salton,[28] ACM representative for the North East region, felt that the resolution had a strong political component and that ACM should not get involved, whereas Thomas D'Auria,[29] ACM representative for greater New York was strongly in favor of ACM doing all they could to assist Soviet computer scientists. Many participants commented on the amount of paperwork involved, over 150 pages, and the fact that half the Council's meeting time was taken up with this issue. Peter Denning,[30] who attended the meeting as ACM Council Member at Large, noted that it would have been more appropriate

26. Andrei Sakharov was a Russian nuclear physicist, Soviet dissident, and human rights activist. He described anti-ballistic missile defense as a major threat of world nuclear war, and as a consequence was banned from conducting any military-related research. In 1970 he was one of the founders of the Committee on Human Rights in the USSR. In 1972 Sakharov became the target of sustained pressure and intimidation from his fellow scientists in the USSR Academy of Sciences, and received direct threats of physical assault.

27. McCracken Papers, Box 5, Folder 7: Turchin Resolution. ACM adopts resolution for dissident soviet scientist. *Computerworld*, (December 10, 1975): 13.

28. McCracken Papers, Box 5, Folder 7: Turchin Resolution. To the ACM Members of the North East Region, Gerard Salton. November 5, 1975.

29. McCracken Papers, Box 5, Folder 7: Turchin Resolution. To the ACM Members of the Greater New York Region, Thomas A. D'Auria, December 1975.

30. McCracken Papers, Box 3, Folder 21: ACM Correspondence on Computers and Public Policy 1974–1977, 1979–80, Letter from Peter Denning May 3, 1976.

for the Turchin issue to take up two minutes of the Executive Council's time, rather than two hours, thus distracting from the ability of the board to conduct relevant scientific and educational business.

The Executive Council first debated whether it should even consider this issue; when that vote was passed,[31] it then considered whether passing a resolution on this issue would be an appropriate action. This debate brought up such concerns as the 1969 Question of Importance vote, the ACM's 501 tax status, the legitimacy of Turchin's membership in the ACM, the motivations behind the offer of a visiting scholarship at Columbia, and the likely effectiveness of any ACM resolution in helping Turchin. After lengthy debate a resolution was passed:[32] "expressing the hope that Dr. Turchin will be permitted to accept the invitation by Columbia University and voicing its concern that he may be prevented from doing so." Gerry Salton commented afterwards that those who opposed the motion were treated as callous and reactionary. His opinion was that ACM had expertise in the educational, scientific, and technical aspects of the computing field and should stick to such matters.[33] Salton was notorious for his colorful accounts of ACM Executive Council meetings, as Peter Denning commented: "We all loved waiting for Jerry's (sic) newsletter to tell what had happened at the Council meeting because it always seemed that Jerry went to a different meeting than the rest of us. His meetings were always more interesting than ours."[34] In contrast to Salton, Thomas D'Auria, held a very different view: "We had an opportunity to help a fellow human being, who is, incidentally, a computer professional. We rose to that occasion."[35]

The opinions expressed by D'Auria led to an exchange of letters between him and Peter Denning[36] which clearly illustrate the different factions within ACM. D'Auria

31. 16 in favor, 5 against, 3 abstaining as reported by Thomas D'Auria, in McCracken Papers, Box 5, Folder 7: Turchin Resolution. To the ACM Members of the Greater New York Region, Thomas A. D'Auria. December 1975.

32. 18 in favor, 0 opposed, 5 abstaining as reported by Gerard Salton in McCracken Papers, Box 5, Folder 7: Turchin Resolution. To the ACM Members of the North East Region, Gerard Salton. November 5, 1975.

33. McCracken Papers, Box 5, Folder 7: Turchin Resolution. To the ACM Members of the North East Region, Gerard Salton. November 5, 1975.

34. Denning, Oral History Interview No. 20, p. 63.

35. McCracken Papers, Box 5, Folder 7: Turchin Resolution. To the ACM Members of the Greater New York Region, Thomas A. D'Auria. December 1975.

36. McCracken papers, Box 3 Folder 21: Peter Denning to Thomas A. D'Auria. 23 December 1975; Thomas A. D'Auria to Peter Denning. January 5, 1976; Peter Denning to Thomas A. D'Auria. January 13, 1976.

was clearly in favor of the association involving itself with the political issues of the day, as a moral duty, while Denning represented the more conservative members who argued that ACM should only comment publically on technical issues that were clearly within its mandate. Denning felt that D'Auria was expressing his personal views rather than reporting impartially on the meeting and that his members in the Greater New York Region should be given the benefit of hearing opposing viewpoints. In Denning's opinion the issue was clearly political, not apolitical as D'Auria suggested. Richard Nance, chair of the External Activities Board, identified the central issue at stake for ACM in the Turchin affair, which was the need to distil the issue of academic freedom into something that is workable within a professional society.[37]

Copies of the resolution were sent to various Soviet scientific officials as well as the chairman of the Communist Party, Leonid Brezhnev, and the Soviet ambassador to the U.S., Anatoly Dobrynin. *Computerworld* noted that ACM had received no reaction from any of these people and that the organization did not plan any further action on the matter. The comment that ACM did not plan to do anything else on the Turchin issue was picked up on by Dan McCracken who expressed his displeasure in a letter to ACM president Jean Sammet;[38] he noted that he had written a letter to *Computerworld* himself, but they had decided not to print it. Turchin was not given permission to take up the position at Columbia, but did immigrate to the U.S. in 1977 and joined the faculty of City University of New York in 1979.

The barriers for Soviet scientists to visit the U.S. was an ongoing issue during the 1970s, and even though ACM took no further action on the Turchin case it did continue involvement with the plight of Soviet computer scientists. In 1976, science faculty at the University of Maryland refused to work with visiting lecturers chosen by the Soviet government until other Soviet researchers who had been denied permission to visit, such as physicist Professor Benjamin Levin, were allowed in.[39] One of the Maryland science staff involved was Jack Minker, chair of the Department of Computer Science, who went on to become vice-chair of the ACM Committee on

37. McCracken papers, Box 3 Folder 21: ACM Correspondence on Computers and Public Policy 1974–1977, 1979–80, Memo from Richard E. Nance. September 13, 1976; J. Minker. 1981. Computer scientists whose scientific freedom and human rights have been violated: A report of the ACM Committee on Scientific Freedom and Human Rights. *Commun. ACM*, 24(3).

38. McCracken Papers, Box 5, Folder 7: Turchin Resolution, Daniel McCracken to Jean Sammet. January 13, 1976.

39. McCracken Papers, Box 3, Folder 21: ACM Correspondence on Computers and Public Policy 1974–1977, 1979–80, Letter from Peter Denning. May 3, 1976.

Scientific Freedom and Human Rights in 1980. He worked together with Anthony Ralston, chair of the committee, who had been lobbying for more ACM activity in the field of human rights for many years. The Committee produced its first major report in 1981,[40] and Ralston served as chair for a further eight years. In later years Ralston reflected: "One of the things I did in ACM, one of the best things I think, was to establish the Committee on Scientific Freedom and Human Rights."[41] Ralston became involved with this issue through his membership of the Committee of Concerned Scientists in Manhattan, and went to Moscow in 1980 with this group to meet refusnik scientists. As he explained: "It was something I felt strongly about personally, and it was just at the time when computer scientists like all other scientists in the Soviet Union were in big trouble, and it was a very good time to try and lend support to them."[42]

6.5 The Committee on Computers and Public Policy

Despite the decisive defeat of the 1969 Question of Importance, a few individual ACM members, most notably Daniel McCracken, continued to press the ACM to engage with political and social issues. One outcome was the establishment of the ACM Committee on Computers and Public Policy (CCPP) in 1973, with Daniel McCracken as inaugural chairman.[43]

One of the first tasks of the new committee was to develop a list of issues concerning computers and public policy.[44] The committee ensured that the topics were widely debated within ACM by working closely with the Special Interest Group on Computers and Society (SIGCAS) to organize discussion panels at the national ACM conferences in 1974 and 1975. At the 1974 conference 15 such topics included information utility for home use, computers and money, computer literacy, computers and privacy, and computers and social power. The issues to be discussed were wide-

40. Minker, Computer scientists whose scientific freedom and human rights have been violated.

41. A. Ralston. 2006. Oral history interview no. 6, by William Aspray and Bernard de Neumann, p. 21. ACM Digital Library, New York.

42. Ibid.

43. Early members of the Committee on Computers and Public Policy included Daniel McCracken (chair), Paul Armer, Robert Ashenhurst, Herbert S. Bright, Jerome A. Feldman, Roy N. Freed, John King, Robert Kling, Peter Lykos, Susan Nycum, Lee L. Selwyn, Bruce Van Atta, and Joseph Weizenbaum. McCracken, Armer and Weizenbaum were also on the executive committee of Computer Professionals against the ABM; Denning, Oral History Interview No. 20, p. 94.

44. D. D McCracken. 1974. A problem-list of issues concerning computers and public policy. *Commun. ACM*, 17(9): 495.

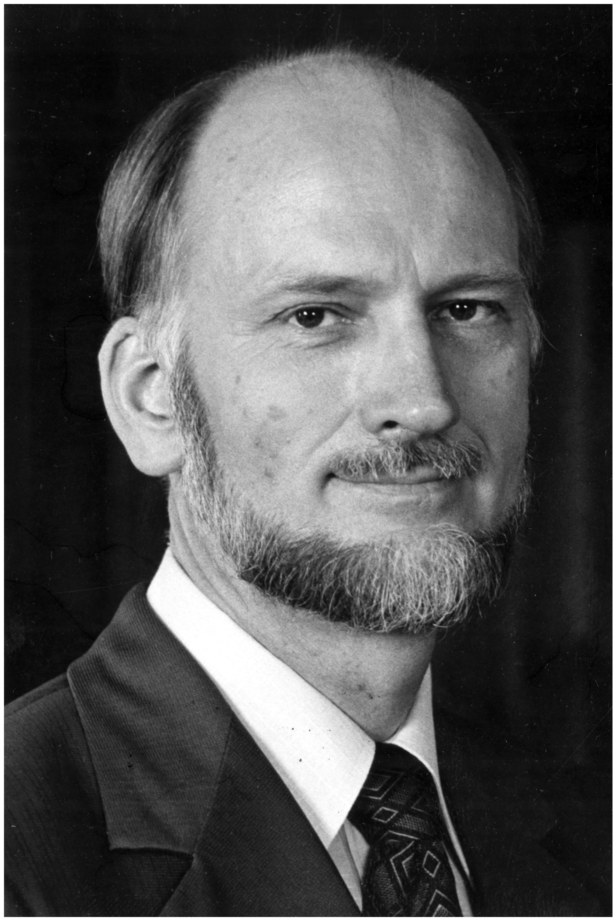

Figure 6.3 **Daniel McCracken, best-selling author, ACM president 1978–1980, activist, and inaugural chair of the Committee on Computers and Public Policy.** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

ranging but all included the use of computing technology as a central issue, an important justification for ACM involvement.

In 1974, ACM president Jean Sammet confirmed McCracken's reappointment as chair of the Committee on Computers and Public Policy, and requested that the Committee should concentrate on the privacy issue.[45] Although this issue was being looked into by the American Federation of Information Processing Societies

45. McCracken Papers, Box 3, Folder 21: ACM Correspondence on Computers and Public Policy 1974–1977, 1979–80, Letter from Jean Sammet. June 24, 1974.

(AFIPS), which had set up a Special Committee on the Right of Privacy headed by Willis Ware, Sammet felt there had been a lack of consultation with ACM. After receiving no reply to her letters to Ware requesting clarification on the participation of ACM in the Privacy Committee[46] Sammet requested McCracken to work independently of AFIPS to develop an ACM paper on the relationship between technology and policy issues. McCracken developed a draft for discussion using the "whimsical" example of someone setting up a lawn mowing business.[47]

Although the 1969 vote on the Question of Importance had been conclusive, just precisely what was and was not a deeply political and social issue was still a matter for debate, and the ruling had also been variously interpreted. In 1975 the Committee on Computers and Public Policy was asked to clarify whether action would be appropriate in individual cases. The guideline developed stated that: "In order for ACM or any of its components to consider commenting or acting upon an issue that could be construed to fall under the deeply political and social questions restriction imposed by the membership in the 1969 Question of Importance vote, the issue must involve a question of specific technical expertise about computers, such that computer people could speak on the subject in a way that members of the general public could not."[48]

Another issue which the CCPP addressed was the human rights of computer scientists in the Soviet Bloc. As previously mentioned, Jack Minker and Anthony Ralston had long been strong advocates for this issue, and when Ralston took over from McCracken as chairman of the CCPP in 1976 the committee redoubled its focus. At first, Minker ran a sub-committee on Scientific Freedom and Human Rights as an offshoot of CCPP, and after lobbying by Ralston this became an ACM standing committee in its own right from 1980 with Ralston as chairman.[49]

From its beginnings the CCPP worked closely with SIGCAS which already had interests in this area having established its own Committee on Information and

46. McCracken Papers, Box 3, Folder 21: ACM Correspondence on Computers and Public Policy 1974–1977, 1979–80, Jean Sammet to Willis Ware. July 25, 1974; Jean Sammet to Daniel McCracken. July 25, 1975.

47. McCracken Papers, Box 3, Folder 21: ACM Correspondence on Computers and Public Policy 1974–1977, 1979–80, Draft: Computer Aspects of the Privacy Issue. September 2, 1974.

48. J. Sammet. 1975. ACM President's letter: Guidelines for deeply political/social issues. *Commun. ACM*, 18(5).

49. McCracken Papers, Box 3, Folder 20: Committee on Computers and Public Policy 1979–80, Letter from Anthony Ralston. October 1, 1979; Note of telegram sent to Anthony Ralston from Paul Armer. February 12, 1980.

Public Policy chaired by Ted Sterling. Several members of SIGCAS were part of McCracken's committee.[50] SIGCAS also organized the session at the ACM '74 conference[51] at which over 150 participants discussed the interim report on "A Problem List of Issues in Computers and Public Policy"[52] for over three hours.[53]

6.6 History of SIGCAS

The website of the Special Interest Group for Computers and Society (SIGCAS) notes that the group addresses the social and ethical consequences of widespread computer usage. Its main goals are to raise awareness about the impact that technology has on society, and to support and advance the efforts of who are involved in this important work.[54] "Our areas of involvement include computer ethics, universal access to computer technology, security, privacy and reliability. We collaborate with other ACM bodies that are engaged in related work such as USACM (ACM US Public Policy Council) SIGITE (SIG Information Technology Education, 2001–2015) and SIGCSE (Special Interest Group on Computer Science Education)."

The first recorded mention of this group was in 1966, when A.W. Holt sent a memo to ACM leadership outlining plans for a new Special Interest Committee on the Social Implications of Computing.[55] The new SIC (sometimes referred to as SIC^2) was envisaged to cover topics such as privacy, the possible consequences of fourth generation languages, and computers in education. Government policies and industrial policies pertinent to the area were to be considered as was the university training of young computer professionals to ensure that they were capable of exercising their responsibilities as regards the social implications of their work.

There was a specific mention of Edmund Berkeley's request to form a subcommittee on "The Use of Computers to the Benefit of Mankind," alongside a note that he would be welcome to do so with "more power to him," as long as he took note of the proviso that he should not undertake actions damaging to the Committee or

50. For example, Robert Ashenhurst, Rob Kling, Peter Lykos in Association for Computing Machinery Records (CBI 205) at http://purl.umn.edu/51982, Box 25, Folder 8, Annual Committee Report. July 10, 1973.

51. ACM Records, Box 25, Folder 8, Annual Committee Report. April 3, 1975; R. Ashenhurst. 1973. Chairman's letter. *Comput. Soc.*, 5(1).

52. McCracken, A problem-list of issues concerning computers and public policy.

53. P. G. Lykos. 1974. Chairman's letter. *Comput. Soc.*, 5(3).

54. SIGCAS. Special Interest Group for Computers and Society. Available at http://www.sigcas.org.

55. ACM Records, Box 25, Folder 8, Plans for SIC2 alias SICSR. September 27, 1966.

the ACM. A 1975 editorial in *Computers and Society*,the SIGCAS newsletter, confirms that SICCAS/SIGCAS was a direct successor of this earlier group.[56]

"SIC^2" had the ambitious aim of building and maintaining a library with an associated reference card distribution service. Early plans outlined how the reference system would work. However the initial launch of the library was delayed, and letters[57] had to be sent to prospective subscribers of the information retrieval service returning their orders and checks. One of the disappointed subscribers was Robert Bigelow who was later to become the first chairman of SICCAS.

At the 1967 Spring Joint Computer Conference SIC^2 ran a program attended by over 100 people on the social problems engendered by computers, but by 1968 the group's future was in doubt. A 1968 ACM reference guide to SIGs/SICs[58] notes that the group, described as: "a forum for discussions related to the social responsibilities of computer professionals" had been dissolved. In 1969 ACM President Bernard Galler announced that he planned to formally dissolve the SIC due to lack of interest. This led to widespread protests from members and a petition was organized by Robert Bigelow to resurrect "the only group in the ACM through which a member can show he cares about the interface of the computer industry with the rest of the world."[59] The petition was signed among others by Paul Armer, who had also been on the executive committee of Computer Professionals against the ABM.

New York-based members of SICSIC wrote a strongly worded article published in *Interrupt,* the newsletter of Computer Professionals for Peace. The authors claimed that the proposed dissolution of SICSIC "effectively demolishes the ACM's pretense of professional neutrality." Chapter 10 also discusses the tensions in the ACM caused by this article, and the subsequent attendance of Computer Professionals for Peace at the 1969 AFIPS Spring Joint Computer Conference. The *Interrupt* article, "On the Social Implications of Computers," ended with the resolution:

> As professionals in the computer field and members of SICSIC, we have the responsibility, through our professional association, to oppose the use of our skills for destructive and anti-social ends. Therefore we urge that ACM adopt these proposals as part of its national policy:

56. C. P. Landis. 1975. Editor's column. *Comput. Soc.*, 6(1).

57. ACM Records, Box 25, Folder 8, Letter from J. D. Madden. September 29, 1967.

58. ACM Records, Box 23, Folder 23, Committee on Special Interest Committees and Groups. July 25, 1969.

59. Berkeley Papers, Box 10, Folder 2, Robert Bigelow to Edmund Berkeley. March 4, 1969.

1. We oppose the war in Vietnam, the U.S. military presence throughout the world, and economic and political subversion of other nations. Since there is widespread involvement of our profession in these endeavors, we urge all computer professionals to review the moral consequences of their involvement in furthering these efforts.
2. We oppose discrimination as practiced in the computer field by direct or indirect means such as educational requirements, arbitrary testing procedures, and restrictive promotion policies.
3. We oppose the establishments of mass data banks which pose a threat to our privacy and concentrate power in the hands of a few.
4. We oppose the economic exploitation of the uninformed by unscrupulous computer schools. We support the implementation of accrediting standards for the computer educational field.
5. We support the active encouragement, development, and funding of programs for the constructive application of computers toward the solution of the many problems faced by our society.[60]

It is not clear whether the controversial views expressed in the article such as "SICSIC and ACM must abandon the misguided concept of professional detachment from political issues" and "It [ACM] should initiate and support political action to end the war" were the views of the group as a whole or a small number of New York members.

In any event the 1969 petition was sent to ACM to retain the Special Interest Committee on Social Implications of Computing with a proviso that the actual name used may be somewhat different.[61] The new SIC was also noted later in the same year by the Committee on Special Interest Committees and Groups.[62] By August the SIC on Computers and Society (SICCAS) was formally created, with a brief for: "Furthering an understanding of how the structure and operation of society may be affected, positively or negatively, by the development and applications of computers and automata."[63] This was a rather less radical than the resolution proposed in

60. Berkeley Papers, Box 10, Folder 2, *Interrupt* Newsletter. March 1969.

61. ACM Records, Box 23, Folder 23, Report to ACM Council on overall SIG/SIC activities and specific action items. April 3, 1969.

62. ACM Records, Box 23, Folder 3, Committee on Special Interest Committees and Groups. July 25, 1969.

63. ACM Records, Box 23, Folder 25, Annual Report of SIG or SIC to ACM Council. August 1970.

the *Interrupt* article. Robert Bigelow was installed as the chairman of SICCAS[64] and began attending ACM SIG Committee meetings on behalf of the group.[65]

By July of the following year, around 290 people had joined the SIC, Michael A. Harrison had been appointed as vice chairman and Sydney Feldman as secretary/treasurer with Grenville Bingham taking on responsibility for producing the group's newsletter, *Computers and Society*, first sent out in November 1969.[66] One of the first panels the SIC held in November 1969 was on the "Education of Non-Computer People,"[67] which included a talk by John Adams from IBM on a project to train undereducated people in Los Angeles to be operators and programmers.[68]

Education was also the topic area for one of the five subcommittees formed after the group's first formal meeting in August 1970, with the other four subcommittees covering the areas of data banks and privacy, jobs in automation, intersociety liaison, and legal aspects.[69]

Because the topics covered by SICCAS were beyond ACM's traditional technical areas, the ACM Executive Committee was concerned that it could be taken over by outsiders. This trepidation was later described rather sarcastically by Charles Davidson, chairman of SIGCAS from 1977–1980: "Because there were apprehensions at the time that a group so orientated could be taken over by wild-eyed radicals and prove an embarrassment to the parent organization."[70] Perhaps there were some grounds for these concerns. The publication of the lengthy and rather radical article on the mandate of SICSIC had been in *Interrupt*, a publication of Computer Professionals for Peace which was co-chaired by Edward Elkind who was an active communist and peace protester.[71] As discussed in Nelsen's chapter, Donn Parker wanted ACM to put controls on such "militant activists."

64. ACM Records, Box 23, Folder 24, Committee on Special Interest Groups and Committees. October 10, 1969.

65. ACM Records, Box 23, Folder 24, Committee on Special Interest Groups and Committees. August 12, 1969.

66. ACM Records, Box 23, Folder 25, Annual Report of SIG or SIC to ACM Council. August 1970.

67. The speakers were Mrs. Barbara Farquhar of Massachusetts General Hospital and Mr. John Adams of IBM.

68. R. P. Bigelow. 1970. From the chairman. *Comput. Soc.*, 1(2).

69. ACM Records, Box 23, Folder 25, Annual Report of SIG or SIC to ACM Council. August 1970.

70. C. H. Davidson. 1979. From the chairman. *Comput. Soc.*, 9(3–4).

71. See Edward Elkind, lifelong peace activist, dies. *PeoplesWorld*, (2011). Available at http://peoplesworld.org/edward-elkind-lifelong-peace-activist-dies/.

The fledgling SIC was allowed only limited self-government. Its management was appointed by ACM rather than elected by the SIC members, and an advisory panel of ACM-approved advisors was established.[72] Another restriction limited the effectiveness of votes by non-ACM members, by making sure they could not contribute more than 50 percent of the total vote. In reality these fears were unfounded and the non-ACM membership of SICCAS remained low.[73] By 1975 the Executive Council[74] felt confident enough to allow SIGCAS to elect its own officers,[75] although the first elections weren't actually held until 1979. It was also apparent, at least in the early years, that many ACM members belonged to multiple SIGs; for instance, in 1975 that more than two thirds of SIGCAS members belonged to at least two other SIGs.[76] This does show the breadth of ACM members' interests at the time, but also raises questions about the strength of their commitment to individual SIGs, especially as they were relatively cheap to join. Of the 916 SIGCAS members in 1975, only 89 (less than 10%) belonged to SIGCAS alone.

At the beginning of 1971, the ACM Council formally approved the group as a SIG[77] and Robert Bigelow asked to be replaced as chairman, noting that it had always been his intention to take the group from SIC to SIG status and then step down.[78] Robert Ashenhurst[79] took over as chairman, and in his first column for *Computers and Society* made the point that SIGCAS was fundamentally different from other SIGs: even though many ACM members were interested in the problems of computers and society, few were professionally occupied in this area.[80]

By 1972 SIGCAS had around 600 members,[81] and next year a merger with SIGSPAC (the special interest group on urban data systems, planning, architecture

72. Paul Armer, Robert Fano, Herbert Grosch, Michael Harrison, Anthony Oettinger, Donn Parker, and Jean Sammet.

73. Davidson, From the chairman.

74. ACM Executive Council Meeting. July 11, 1975.

75. P. G. Lykos. 1975. Chairman's column. *Comput. Soc.*, 6(3).

76. Ibid., 6(2).

77. SIC (Special Interest Committee) generally was formed first to determine whether there was enough interest to support activities in that technical area. If this so, usually after a period of around a year, the SIC converted to a SIG (Special Interest Group) which is financially self-supporting, through member dues, and with its own officers.

78. ACM Records, Box 23, Folder 26, Annual Report of SIG or SIC to ACM Council. August 1971.

79. Robert Ashenhurst's ACM papers are deposited at CBI but are not yet (2016) catalogued.

80. R. Ashenhurst. 1972. From the chairman. *Comput. Soc.*, 3(3).

81. Ibid., 3(4).

and civil engineering) increased the membership to over 1,000.[82] Between 1974 and 1978 total membership rose from 973 to 1,069. However in ACM terms SIGCAS has always been one of the smaller special interest groups. The 1985 Board report noted that SIGCAS was classified as a "very small" SIG with under 1,500 members[83] and by comparison SIGCSE was classed as "small" with between 1,500 and 3,200 members, whereas the three largest SIGs—SIGGRAPH, SIGPLAN and SIGSOFT[84]—had over 10,000 members and significant financial resources. Although there were issues around how SIGs were funded, with ACM starting to charge SIGs for services provided centrally and placing restrictions on external bank accounts, these matters were not particularly relevant to SIGCAS. Despite its relatively small size there were never any doubts about its ongoing viability. All SIGs were reviewed on a regular basis, and a 1981 review of SIGCAS confirmed that it continued to be an active group, with a loyal member base and regular newsletters.[85]

Beginning in 1970 SIGCAS published a quarterly newsletter,[86] *Computers and Society*, which contained non-refereed articles, SIG news, book reviews, and letters. In 1973, when Carolyn Landis was appointed as associate editor, the newsletter became much more formal in style, with three associate editors being established for Human Impact (Rob Kling), Computers and Government (James Oliver), and Information and Policy, Public and Private (Theodor Sterling).[87] The newsletter published articles by many researchers who went on to become highly influential in the field, such as Rob Kling himself, Allan Westin, Murray Turoff, and Starr Roxanne Hiltz.[88]

Computers and Society contained controversial material from the start, with an early issue including a letter from group secretary/treasurer Sydney Feldman[89]

82. Note this claim for over 1000 members in the editorial is not supported by the actual data submitted to ACM; C. P. Landis. 1975. Editor's column. *Comput. Soc.*, 6(1).

83. ACM Records, Box 24, Folder 10, SIG Board Report. March 7, 1985.

84. For example, SIGGRAPH had a total budget of US $696,969 as compared to US $43,462 for SIGCSE and US $19,625 for SIGCAS.

85. ACM Records, Box 24, Folder 6, SIG Board Actions at NCC '81. May 20, 1981; C. H. Davidson. 1978. From the chair. *Comput. Soc.*, 9(2).

86. In some years there have been gaps in the production of issues.

87. G. R. S. Bingham. 1973. Editor's letter. *Comput. Soc.*, 5(1).

88. For example, as of September 29, 2015, Rob Kling had many articles with over 500 citations, Alan Westin's 1970 paper on Privacy and Freedom had been cited 3,327 times, Murray Turoff had two books with over 7,000 citations, Starr Roxanne Hiltz had two books with over 3,000 citations. (Google Scholar)

89. S. Feldman. 1970 Letters. *Comput. Soc.*, 1(2).

debating the relationship of computer professionals to the war in Vietnam, plus coverage of Edmund Berkeley's provocative address to the ACM Silver Banquet Committee on the occasion of the acceptance of his lifetime award.[90] In that speech, Berkeley brushed off SIGCAS as an "example of tokenism." Other SIG members also felt that the group was not radical enough. Richard Sprague wrote a long letter[91] expressing his disappointment in response to an editorial in the summer 1974 issue trying to encourage more reader input.[92] His letter suggested that the assassination of John F. Kennedy should be reinvestigated using newly available computer evidence, and expressed his concerns about the clandestine use of surveillance by the intelligence community.[93] Berkeley, as the editor of *Computers and Automation*, was apparently one of the few people who had taken his claims seriously. Sprague offered three controversial topics for discussion by SIGCAS members.

> How can computing and other technology be used to discover who is assassinating our leaders and why?
>
> What technology-based activities are leading us toward Technofacism and how can they be exposed and stopped?
>
> Where and how are agencies of the U.S. government using technology around the world to help suppress people in other countries?

Carolyn Landis responded by pointing out that SIGCAS was not a plenary body where issues could be considered and resolved, meaning that *Computers and Society* could not represent the opinion of the whole group;[94] what it could do was act as a forum for the expression of opinions on computing-related social issues by individual members. In the following issue, James Oliver, the associate editor of the Computer and Government section, noted that Sprague's challenge was being met and invited readers to send in their articles about controversial topics—with the caveat that these should be of professional quality and not sensational.[95]

Questions about SIGCAS were also raised by its younger members. The student chapter at University of Missouri–Rolla challenged chairman Peter Lykos, when

90. Ashenhurst, From the chairman; B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals*. Association for Computing Machinery and Morgan & Claypool, New York.

91. R. E. Sprague, Issues for SIGCAS. *Comput. Soc.*, 5(3).

92. C. P. Landis. 1974. Editor's letter. *Comput. Soc.*, 5(2).

93. He mentions that he is a member of "The Organizing Committee for the Fifth Estate."

94. C. P. Landis. 1974. Editor's column. *Comput. Soc.*, 5(3).

95. J. R. Oliver. 1974. Computers and government section. *Comput. Soc.*, 5(4).

he gave a talk there, noting that there seemed to be lots of discussion and debate within the group but very little action, a challenge that he reported in turn to SIGCAS members.[96]

As previously mentioned, SIGCAS had a close relationship with the Committee on Computers and Public Policy, but in other areas there was a lack of co-operation between the different SIGs and even some suggestion of "turf wars." ACM received a National Science Foundation (NSF) grant of $34,000 to support "A Study of Computer Impact on Society and Computer Literacy Courses and Materials" to be run by Richard Austing and Gerald Engel of the special interest group on computer science education (SIGCSE). The chairman reported that even though SIGCAS was the one special interest group that specifically addressed those issues, it had not been consulted in any way.[97] In the same issue the chairman announced that ACM had requested SIGCAS to create guidelines regarding the confidentiality of ACM databases, especially with regard to the membership profile and opinion survey.[98] The wider ACM community evidently viewed SIGCAS as a group that was mainly concerned with privacy issues, rather than developing computer literacy or addressing social issues.

Relations between SIGCAS and ACM management were generally positive. In 1976 the SIGCAS chairman Ellis Horowitz noted[99] that both the new president and vice president of ACM (Herb Grosch and Dan McCracken, respectively) were committed supporters of SIGCAS, and both were strongly in favor of ACM exploring the social implications of new technology.

Individual SIGCAS members sometimes used their enthusiasm for a cause to influence the rest of the group. For example, in 1975 Edward Robertson was appointed as vice chairman; he had recently returned from a year teaching in Ghana and had a special interest in the use of computers in developing nations. He organized a panel on the topic at ACM '76 and published an associated article in the newsletter.[100] Other individuals used SIGCAS to reach the wider ACM community. In 1975, the *Communications of the ACM* added a new column on the social implications of computing headed by Rob Kling.[101] In his first column Kling noted: "As

96. P. G. Lykos. 1974. Chairman's column. *Comput. Soc.*, 5(4).

97. P. G. Lykos. 1974. Chairman's letter. *Comput. Soc.*, 5(3).

98. Ibid.

99. E. Horowitz. 1976. Chairman's column. *Comput. Soc.*, 7(2).

100. E. L. Robertson, B. W. Boehm, H. D. Huskey, A. B. Kamman, and M. R. Lackner. 1976. Computers in developing nations. *Comput. Soc.*, 7(2): 7–9.

101. E. Horowitz. 1975. Chairman's column. *Comput. Soc.*, 6(4).

computing has become more pervasive, social issues raised by technology have become more important."[102] He commented that two of the earliest concerns related to computers were threats to privacy and loss of jobs, while loss of employment was not considered such a pressing issue in 1976; privacy remained a major concern.

6.7 Analysis of Changing Interests within SIGCAS

During the period between its establishment and 1985, the focus of interests within SIGCAS was changing, influenced by developments in the outside world and within ACM. For SIGCAS's 40th anniversary Joseph Oldham looked back over the articles that had been published in *Computers and Society*,[103] and developed an ad hoc taxonomy as shown in Appendix A as Table 6.1. He categorized 371 articles in total,[104] with the majority (around 80%) having been written by academics. He noted that his analysis confirmed that the group has always been a diverse community, and that education (EDU) is the second most prevalent topic after the rather general catch-all topic of social interplay between computers and society (SIG). Papers on education have covered such areas as teaching social issues and ethics in computing education, computer literacy, and community education and outreach.

His findings confirmed earlier observations about the history of SIGCAS made by Ronald Anderson[105] who noted that rather than coming about because of general concerns about the effects of computerization on society (as the resolution in the *Interrupt* article might suggest), the main driver for the group was a specific concern about computer-related privacy.

This is certainly backed up by a study of the earlier issues of the newsletter. The first chairman of SICCAS, Robert Bigelow, was a practicing Boston attorney with a strong interest in privacy issues.[106] In the early 1960s the public was beginning to think seriously about computer related privacy issues for the first time as a result of the Kennedy and Johnson administrations floating proposals for a National Data Bank. The number of privacy related hearings by government and the number of articles published about computer-related privacy both peaked in the late 1960s, immediately before the formation of SICCAS.[107]

102. R. Kling. 1976. Social impacts of computing. *Comput. Soc.*, 7(2).

103. J. D. Oldham. 2009. Examining computers and society 1970–2008. *Comput. Soc.*, 39(2).

104. Articles from two special issues in volume 20 no. 3 (1990) that included papers from the Computers and the Quality of Life conference are excluded.

105. R. E. Anderson. 1999. SIGCAS history: The early years. *Comput. Soc.*, 29(4).

106. Ibid., 4–5.

107. Ibid.

Oldham's classification scheme was used to categorize further recorded information which helps to explain the development of the SIG. This information included a 1977 survey of SIGCAS members' research interests shown in Appendix A Table 6.2 and SIGCAS-sponsored conference panels at the national ACM conference between 1973 and 1985 in Appendix A Table 6.3.

In 1977 Elihu Gerson reported on the results of a survey of the research interests of SIGCAS members[108] summarized in Appendix A Table 6.2. Despite repeated calls for participation the response was rather disappointing with only 34 responses. The majority (24) were academics with five respondents from practice, two from professional societies, one from an academic journal, and two not providing any affiliation. As far as could be established from names and titles, only one of the researchers was female, Starr Roxanne Hiltz.[109] The research interests reported are shown in Table 6.2 and have been classified using the descriptors used by Oldham.[110]

This survey of SIGCIS researchers' interests confirms the importance of privacy. Policy related issues, professionalism and ethics, and the broad category of the social interplay between computers and society were also well represented. Education ranked joint 6th in terms of research interests with only four researchers active in this area. Even though educational issues were clearly important to SIGCIS as a group, in 1977 in terms of research interests they were not as significant as other topics, most notably privacy.

During the period from 1973–1985, SIGCAS organized panels for the ACM national conference, averaging about two per conference; some consisted of a number of presentations with formal papers, others of a more informal discussion. These panels also indicate the changing interests of the group over time, and are included as Appendix A Table 6.3 along with the appropriate descriptor from the Oldham classification.

The analysis of these panels shows a different pattern: if education (EDU) is combined with access to computing for the general population (PAC) it actually becomes the first ranked topic area forming the basis of seven out of 28 panels whereas privacy (PRI) is the topic area for only one panel. Other significant areas of interest are policy (POL) and issues in the workplace (WRK) with five panels apiece. However, the policy panels are very much clustered around 1974 and 1975 when the

108. E. M. Gerson. 1977. Directory of researchers in computers and society. *Comput. Soc.*, 8(4).

109. R. Subramanian and T. Haigh. 2013. Starr Roxanne Hiltz: Pioneer digital sociologist. *IEEE Ann. Hist. Comput.*, 35(1): 78–85.

110. Oldham, Examining computers and society 1970–2008.

Committee on Computers and Public Policy was working (as noted above) on the Problem List of Issues in Computers and Public Policy. The educational related panels are more evenly spread out, at least until 1982 when they are overtaken in popularity by the work-related issues panels which tended to focus on more futuristic topics such as the use of robots in the workplace. Appendix A Table 6.3 does seem to provide evidence of a strong ongoing interest in education-related issues within SIGCAS. Education is the second most popular topic for articles in the SIGCAS newsletter, *Computers and Society*, and is also frequently discussed in its front and back matter. For example, a note in the back matter of a 1977 issue of the newsletter reports on a University of Notre Dame program to stimulate talented minority students' interest in technical careers.[111]

6.8 Conclusion

During these two decades, computing as a new discipline was still negotiating its relationship with the wider world. ACM played a crucial role in determining the standing of computer science within American society. Individual ACM members held very different views about what was appropriate for the organization to publically comment on. There were those who would have liked the ACM to be very active in lobbying government on the issues of the day, and others with a more conservative approach who felt that any public opinions expressed by the organization should be strictly limited to technical matters that were clearly central to the discipline.

During the 1960s and 1970s controversial issues such as the Vietnam War and the Equal Rights Amendment brought some of these tensions to a head, and the examples used in this chapter have given a flavor of the discourse that arose around these issues as ACM negotiated its public image. The 1969 Question of Importance vote was a watershed moment, in that it provided a very clear indication that the overwhelming majority of ACM members were *not* comfortable with the organization becoming involved with deeply political and social issues such as use of computer technology in warfare. As different issues were proposed for debate over the years, the Question of Importance vote, together with the specter of the 501 tax status held by ACM, had a strong moderating effect on what matters were deemed suitable for the ACM to take a public position on.

The topics ACM regarded as suitable were those that had a strong computing component at the core. For example, computing related privacy issues such as the

111. Anonymous. 1977. Computers interest minority students. *Comput. Soc.*, 8(3).

use of social security numbers as universal identifiers and the electronic transfer of bank funds were issues that ACM was prepared to take a stand on. Issues related to professional conduct—such as weeding out bogus computer schools and making sure that enough computer scientists were being trained at university level—were areas where ACM felt comfortable with making a contribution. On the other hand, concerns such as the human rights of Soviet computer scientists, the Equal Rights Amendment, and the use of computers in missile defense were deemed to be "political" and caused a lot more internal controversy.

There were groups within ACM where "deeply political and social issues" could be debated, the two most significant being SIGCAS and the Committee for Computers and Public Policy. As Peter Denning saw it, these groups provided an outlet for those ACM members who were interested in such matters and wished to get involved.[112] However, these groups were still part of ACM, and any influence they wanted to have on the outside world had to be negotiated through the organization. In its early days when there were fears of a radical takeover, SIGCAS was quite tightly controlled by the ACM Executive who appointed the group's officers. In later years, when the group had proved itself to be "safe," members were allowed to elect their own officers. Similarly the topics that the Committee on Computers and Public Policy investigated were often dictated by the ACM Executive, such as when ACM President Jean Sammet asked them to concentrate on the privacy issue. SIGCAS on occasion was directed to look at certain issues by the ACM Executive.

For a special interest group that was classed was "very small" the achievements of SIGCAS were significant. For 15 years it consistently organized well-attended panels at national ACM conferences, its members included several internationally well-regarded researchers, and it continued to produce a quarterly newsletter. The group made a significant contribution to the development of tertiary level curricula on computers and society.

Individuals within SIGCAS or CCPP sometimes were able to use their positions to influence the wider organization. For example, Anthony Ralston and Jack Minker's advocacy for the plight of Soviet computer scientists resulted in the establishment of an ACM Standing Committee on Human Rights. The first chairman of SIGCAS, Robert Bigelow, used his position to try to influence policy over privacy issues and in 1975 SIGCAS vice chairman Edward Robertson started a debate about the use of computers in developing nations.

112. McCracken Papers, Box 3, Folder 20: Committee on Computers and Public Policy 1979–1980, ACM and Political & Social Issues, Peter Denning, Council Member at Large. December 23, 1975.

Some areas of protest moved outside ACM. In 1983, Computer Professionals for Social Responsibility (CPSR) was formed in reaction to concerns about the use of computers in nuclear war. The group was set up completely independently of ACM, and founder member Severo Ornstein commented that initially organizations such as ACM and AFIPS didn't want anything to do with them.[113] In the early stages CPSR often encountered the view that "computer people have no social concerns"[114] and "that's how the field was characterized. To be looked at from the outside you didn't see people with social concerns on the inside."[115] However, by the 1990s ACM was working more closely with CPSR, supporting their meetings and organizing debates around different issues.

In general, the ACM concentrated its efforts to influence policy on the areas which were clearly within its area of expertise and where there was a consensus within the organization about the appropriateness of making a contribution. The topic where it made the greatest input during this period was in the area of computer-related privacy, which was an ongoing interest of SIGCAS and CCPP.

In the early days, there were a few key individuals who tried to push the ACM into taking a radical stance on topical issues such as the use of computing technology in the Vietnam War. These views were never felt to be representative of the majority of members and this was put to the test during the 1969 Question of Importance vote, when the more radical element of ACM was soundly defeated. After this event, the ACM became more conservative and focused on less controversial issues that were well within its areas of competence. Various sections of the organization such as SIGCAS and CCPP were allowed to debate deeply social and political issues but their public activities were kept under the control of the ACM Executive. Nevertheless the discourse that did take place had some impact, and individuals within both SIGCAS and CCPP were able to influence the ACM to occasionally take a public stand on some "deeply political and social issues."

113. S. Ornstein and L. Gould. November 17, 1994. Oral History Interview by Bruce Bruemmer, Charles Babbage Institute (OH 258), p. 11. Available at http://purl.umn.edu/107336.

114. Ibid.

115. Ibid.

Appendix A

Table 6.1 Subject categories of papers published in Computers and Society, 1970 to 2008

Descriptor	Interpretation	Occurrences
SIG	Social Interplay (General) between computers and society. 19 papers were marked with only SIG	151
EDU	Educational issues of computing and society, computing ethics etc.	83
SIGi	Social Interplay (General) between computers and society with a significant network. Internet component	82
ETH	Ethics, be it an examination of ethical frameworks or presentation of case studies, to use in the classroom	60
PRI	Privacy	57
SEC	Security and crime	51
POL	Policy	45
LEG	Legal issues	35
PRO	Professionalism for members of the computing industry	34
COM	Issues of commerce, from corporate behavior to interests of the consumer. Issues of economics are included here.	30
WRK	Issues in the workplace	29
FOC	Field of computing: where does computing fit into the world, into funding, how might computing be used in a particular domain such as health care, or by government agencies.	22
IP	Intellectual property	20
RES	Addressing or pertinent to research in computers and society (CAS)	16
ACT	Calls for or discussion of social activism related to computing	12
CHI	Computer Human Interaction	11
PAC	Access to computing resources for general populations (e.g., a paper addressing the Digital Divide)	10
CIV	Civil liberties (including freedom of speech, censorship)	8
HEL	Health care	8
REG	Regulation (of the Internet, or earlier of networks)	7
INT	Internal to SIGCAS	5
AIN	CAS perspectives on Artificial Intelligence	5
BIM	Biometrics	4
DAC	Access to computing for populations needing different computing hardware and software, e.g., haptic interfaces	4
WIC	Women in Computing	4
HUM	Computing in the Arts and Humanities	3
GAM	Gambling and virtual worlds from a CAS perspective	3
LIM	Limits of computing related to CAS issues. For example, using arguments about the limits of computing to address the wisdom of the SDI program.	3
NOC	No category able to be described	3
POR	Issues related to pornography	2
ENV	Issues related to the environment	2

Table 6.2 1977 Survey of research interests of SIGCAS members

Topic	No. Researchers	Descriptor
Privacy issues	10	Privacy (PRI)
Wider impact on society	10	Social interplay between computers and society (SIG)
Social responsibility of computer professionals	9	Professionalism for members of the computing industry (PRO), Ethics (ETH)
Policy issues	5	Policy (POL)
Working environment	5	Issues in the workplace (WRK)
Technology & Social Organization	4	Social interplay between computers and society (SIG)
Impact on Employment	4	Issues in the workplace (WRK)
Computer Science education	4	Educational issues of computing and society (EDU)
Electronic Funds Transfer	3	Issues of commerce (COM)
Computerized conferencing	3	Social interplay between computers and society with a significant network/ Internet component (SIGi)
Computers for development	2	Access to computing resources for general populations (PAC)
Multinational issues	2	Policy (POL), Legal issues (LEG)
Database Management	1	Field of computing (FOC)
Politics of system development	1	Policy (POL)
Qualitative problems	1	Addressing or pertinent to research in computers and society (RES)
Marketing	1	Issues of commerce (COM)
Software workers	1	Professionalism for members of the computing industry (PRO), Issues in the workplace (WRK)
Women in EDP	1	Women in computing (WIC)
Health	1	Health care (HEL)
Networked Information services	1	Social interplay between computers and society with a significant network/ Internet component (SIGi)
Information Utilities	1	Issues of commerce (COM)
Ethics of machine intelligence	1	CAS perspectives on Artificial Intelligence (AIN)
No interest stated	2	No category (NOC)

Table 6.3 SIGCAS Panels submitted to ACM Annual Conference 1973–1985

Year	SIG/SIC	Topic	Descriptor
1969		Computing and the Disadvantaged	Access to computing resources for general populations (PAC)
1973	SIGCAS & SIGSOC	Social Research on Computer Impact	Social interplay between computers and society (SIG)
	SIGCAS	Computer Impact on Society and Vice Versa	Social interplay between computers and society (SIG)
1974	SIGCAS & SIGCSE	Courses on "Computers and Society"	Educational issues of computing and society (EDU)
	SIGCAS	Forum on Computers and Public Policy	Policy (POL)
1975	SIGCAS 1	Information and Public Policy	Policy (POL)
	SIGCAS 2	Forum for ACM Committee on Computers and Public Policy	Policy (POL)
	SIGCAS 3	Computers in the Electoral Process	Policy (POL)
	SIGCUE 1 (with SIGCAS & SIGCUE)	Professional Societies Focusing on Computers in education	Educational issues of computing and society (EDU)
1976	SIGCAS 1	Computers in Developing Nations	Access to computing resources for general populations (PAC)
	SIGCAS 2	Prospects and Problems with Electronic Funds Transfer Technologies	Issues of commerce (COM)
1978	SIGCAS	How Fast Should EFT be Moving	Issues of commerce (COM)
	SIGCAS/ SIGCSE	Report on the ACM/NSF Workshop on Computers and Society and Computer Literacy	Social interplay between computers and society (SIG) Educational issues of computing and society (EDU)
	SIGCAS	Research in Computers & Society 1	Social interplay between computers and society (SIG) (PRI) (SEC)
	SIGCAS	Research in Computers & Society 2	Social interplay between computers and society (SIG) (CHI) (POL) (ETH))
1979	SIGCAS[a]	The ACM Project on Computers and Society and Computer Literacy Courses and Materials	Educational issues of computing and society (EDU)

a. *Denotes assumption that panel is SIGCAS.*

Table 6.3 *(continued)*

Year	SIG/SIC	Topic	Descriptor
1981	SIGCAS[a]	Robots and their impact on employment	Issues in the workplace (WRK)
	SIGCAS[a]	Elementary Computer Literacy	Educational issues of computing and society (EDU)
1982	SIGCAS	What tasks should robots perform? A social planning issue	Issues in the workplace (WRK)
1984	SIGCAS[a]	Wealth and Jobs in the Fifth New World	Issues in the workplace (WRK)
1984	SIGCAS[a]	Ethical Issues in New Computing Technologies	Ethics (ETH)
1984	SIGCAS[a]	Social and Organizational Consequences of New Generation Technology?	Social interplay between computers and society (SIG)
1984	SIGCAS[a]	Workplace Impacts—AI and Automation	CAS perspectives on Artificial Intelligence (AIN)
1984	SIGCAS[a]	Privacy and Accountability of Large-Scale High-Capacity Personal Data Systems	Privacy (PRI)
1984	SIGCAS[a]	Social Dimensions of Reliability of Complex systems	Security and crime (SEC)
1984	SIGCAS[a]	Social Implications of Artificial Intelligence	CAS perspectives on Artificial Intelligence (AIN)
1985	SIGCAS	SHI1 Computers in the Workplace	Issues in the workplace (WRK)
1985	SIGCAS	SH2 Computers and the Future of Work	Issues in the workplace (WRK)

a. *Denotes assumption that panel is SIGCAS.*

7

Organized Advocacy for Professional Women in Computing: Comparing Histories of the AWC and ACM–W

Amy Sue Bix

7.1 Introduction

Technical work and engineering in the U.S. have always reflected social, cultural, and psychological assumptions about men, women, knowledge, and appropriate employment. But that contested history has not remained static; to the contrary, over recent decades, it has shifted dramatically in tone. While women had a significant presence in computing during and immediately after World War II, top positions in the field went overwhelmingly to men throughout the late 20th century. Few observers in the worlds of business, academia, or engineering sought to alter that trend in any systematic fashion. Yet by 2016, the American president, leading scientific and engineering bodies, major educational institutions and nonprofit groups, parents, teachers, and women's organizations all frequently voiced concern about female underrepresentation in computers, technology, and science fields. Educational and extra-curricular programs prioritized drawing more young women into computers and engineering, an aim backed by major funding and high-profile endorsements. Women working in the computer world, together with male allies, made fairness a non-negotiable priority and organized efforts to address disturbing inequities. Discussions of technical work, gender, and discrimination attracted national attention in the early 21st century, a new phenomenon in American society.

This chapter examines the history of two organizations, the Association for Women in Computing (AWC) during the late 1970s and early 1980s, and the Association for Computing Machinery's Committee on Women in Computing (ACM-W), which began work in the 1990s and was elevated to ACM Council status by 2009 as an official voice for women's interests and with access to central leadership deliberations. While AWC and ACM-W differed in key ways (including the type of computer work central to many members), participants usually shared a philosophical and personal dedication to promoting the cause of women in computing. But more crucially, comparing the two groups' activities in different decades illustrates how radically the last several decades revolutionized American advocacy for women in computing, giving such efforts new directions, enlarged scope, and vastly more social power.

Computing work had never been reserved exclusively for men. Indeed, as Jennifer Light has detailed, a small group of women played an essential role in developing and running the ENIAC machine at the University of Pennsylvania after World War II. Wartime manpower shortages had earlier led the military to hire mathematically adept women as human "computers" to calculate ballistics tables, making it seem logical to recruit six to program newer electronic devices. Historians have since described Kathleen McNulty, Frances Bilas, and their female colleagues as the country's first computer programmers. But precisely because their work could be misunderstood as akin to typically female clerical labor, the project ultimately reinforced a message that the real importance lay in contributions of male engineers who invented the hardware and issued instructions to the secondary level of female service workers. The "ENIAC girls" (as they came to be nicknamed) quickly developed a thorough knowledge of the equipment that proved invaluable in setting up the machine's operations and diagnosing trouble-spots. Yet such skills lacked glamor, suggesting that these women were merely translating men's technical visions into applied form through routine. Later media coverage rendered invisible these women's contributions to ENIAC's success, literally cropping them out of computer room photographs, while lauding leaders John Mauchly and J. Presper Eckert.[1]

Recent historical work has sought to analyze the gendered patterns of various forms of computer work from the postwar years onward, as increasing adoption

1. J. Light. 1999. When computers were women. *Technol. Cult.*, 40(3): 455–483. Available at http://www.jstor.org/stable/25147356.

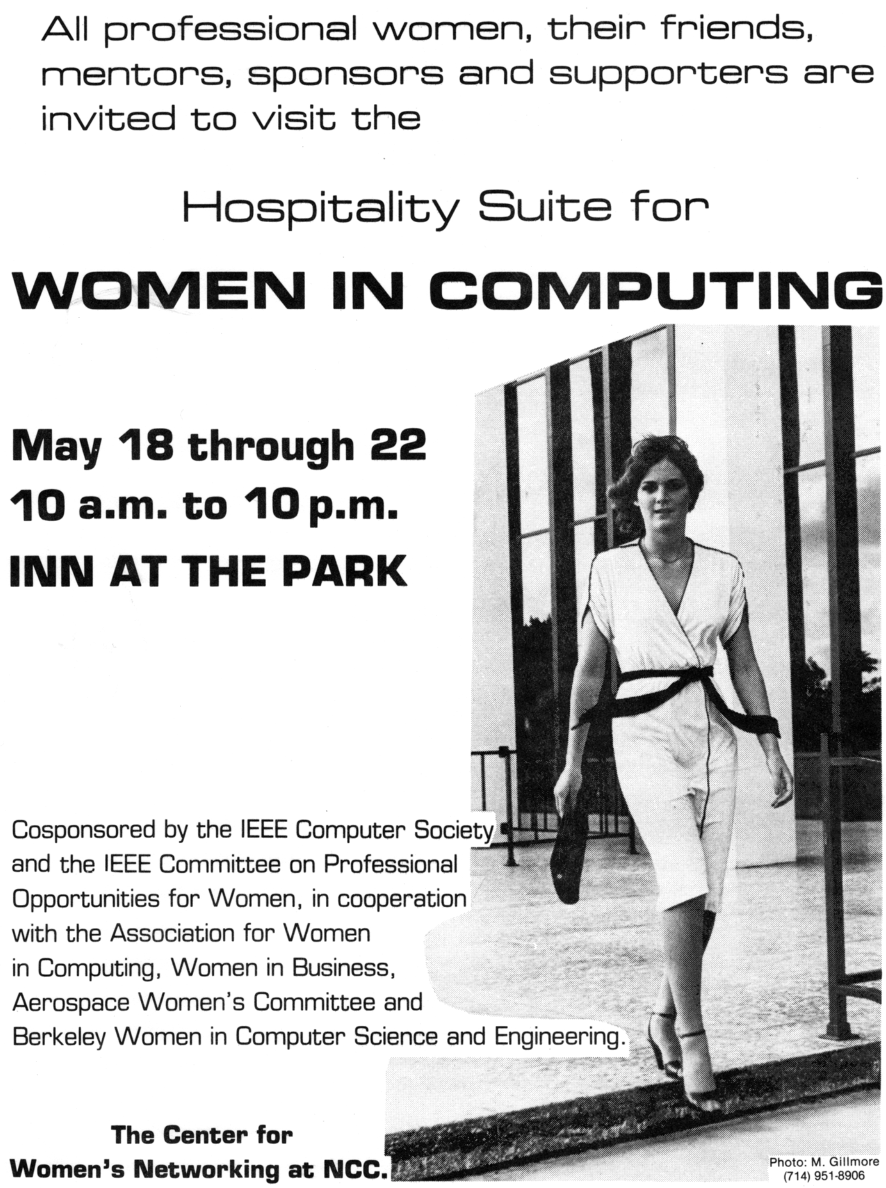

Figure 7.1 **Association for Women in Computing's co-sponsored hospitality suite (1980) illustrates the 1970s emergence of multiple advocacy and support organizations.** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

of the technology in business, government, and universities opened up new jobs. Nathan Ensmenger suggests that women comprised perhaps 30% or even 50% of programming workers through the 1960s, sometimes favored by employers for their supposed patience, dependability, and tractability. Yet at the same time, Ensmenger writes, "as computer programmers constructed a professional identity for themselves during the crucial decades of the 1950s and 1960s, . . . they also constructed a gender identity . . . to distance their profession from its low-status origins in clerical data processing." The "computer boys" (as they came to be known) promoted the idea of programming as a mysterious "black art," linked to "particularly male notions of mastery, creativity, and autonomy." In "a deliberate self-construction embraced by the community," Ensmenger argues, industry personality profiles reinforced the masculinized image of programmers as socially detached, unkempt, and eccentric brains.[2]

But given how quickly hardware options, software development, and users' decisions about computer adoption evolved during each decade of the late twentieth-century, it is no surprise that the gender dimensions of the field also kept shifting. The share of undergraduate computer-science degrees awarded to women in the U.S. *rose* steadily from the start of the 1970s to peak in 1984, when 12,066 out of 32,435 computer-science bachelor's degrees (37%) went to female graduates. From then onward, women's proportion of the field dropped, falling almost as fast as it had grown. From 2008–2012, women claimed just 18% of the nation's computer-science bachelor's degrees.[3]

Looking beyond degree counts, the changes in computer jobs pose more knotty challenges for historical interpretation, precisely because information handling in business, government, academic, and other public settings underwent dramatic transformations. The cascading technological shifts—away from punch-card tabulators and enormous mainframes, to workstations, smaller office-oriented mini-computers, word-processors, and personal computers, into the networked age of multimedia, pocket computing-power, and digital ubiquity—entailed complex changes in workplace organization, all with implications for the gendered labor market. Janet Abbate, Marie Hicks, and other historians of computing have docu-

2. N. Ensmenger. 2012. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*, pp. 77, 79, 239–240. MIT Press, Cambridge, MA. See also N. Ensmenger. 2015. 'Beards, sandals, and other signs of rugged individualism': Masculine culture within the computing professions. *Osiris*, 30(1): 38–65.

3. National Science Foundation, Science and Engineering Degrees: 1966–2012. 2015. Table 33. Available at http://www.nsf.gov/statistics/2015/nsf15326/#chp2 (accessed January 2016).

mented women's often-overlooked contributions to the field, as well as the multiple ways that concepts of identity, masculinity, and femininity played into female success, marginalization, or frustrations with technical work.[4]

Writing in 1994, Rosemary Wright and Jerry Jacobs described computer programming and similarly skilled work as "a rapidly feminizing occupation." Drawing on data from the U.S. Bureau of Labor Statistics (BLS), the two sociologists concluded that the category of computer programmers had gone from 23% female in 1971 to peak at about 36% in 1987, then dropping slightly to 34% female in 1991. Over the same 20 years, women's representation among operations systems researchers and analysts had moved from 9% in 1971 to 43% in 1991, while female representation among computer systems analysts and similar specialists had risen from less than 10% in 1971 to over 30% in 1991.[5] In a later look at BLS data, Caroline Hayes suggested that women's representation in software developers had peaked at 42% in 1987, declining to less than 25% in 2006. For systems analysts, Hayes saw a 1989 peak at 38%, falling to 28% by 2006. Drawing on statistics from the Current Population Survey, Thomas Haigh suggests a slightly higher representation of women in programming from 1982–1992 and emphasizes that beyond the ups and downs of female percentages, the era brought such an overall computer-sector surge that the *total* of women in computer work kept growing. "[F]rom 1992 to 2002, aggregate female employment within the categories of software developer and systems analyst/computer scientist increased by 83 percent. . . . [T]he aggregate number of women reported across the high-status categories of computer and information science manager, computer scientist/systems analyst, computer programmer, and computer software engineer rose by 9 percent from 2003 to 2006."[6]

Behind such complex data on the post-1970s high-tech environment lie even more complex questions about how to explain these trends, particularly the radical plunge in women's computer-science degrees between the mid-1980s and the present. Historian Tom Misa declares, "This lopsided change in computing's

4. J. Abbate. 2012. *Recoding Gender: Women's Changing Participation in Computing*, p. 2. MIT Press, Cambridge, MA; M. Hicks. 2010. Only the clothes changed: Women operators in British computing and advertising, 1950–1970. *IEEE Ann. Hist. Comput.*, 32(2): 2–14; T. J. Misa, editor. 2010. *Gender Codes: Why Women Are Leaving Computing*, pp. 3–23. Wiley IEEE Computer Society, Hoboken, NJ.

5. R. Wright and J. A. Jacobs. 1994. Male flight from computer work: A new look at occupational resegregation and ghettoization. *Am. Sociolog. Rev.*, 59(4): 511–536.

6. C. C. Hayes, Computer science: The incredible shrinking woman. In T. J. Misa, editor, *Gender Codes*, 25–49, see Figure 2.3; T. Haigh, Masculinity and the machine man. In Misa, editor, *Gender Codes*, 51–71.

gender balance in the past two decades is entirely without historical precedent."[7] Historians of computing have joined social scientists, educators, and computer specialists themselves in trying to understand the interacting forces of early childhood, undergraduate, and graduate studies, and the complicated context of employment, family, and gendered society, all potentially influencing decisions about computer work. Reviewing recent literature on factors that can attract or deter young women from computer interests, Lecia Barker and William Aspray note analyses of socialization and gender identity, exposure to computing at home and school, gaming and extracurricular activities, early choices of elective subjects and career goals, girls' and boys' confidence with and attitudes toward technology, peer pressure, popular-culture messages about gender and computing, and the effects of family, counselors, and teachers.[8] The subject of women and gender in postsecondary computer contexts deserved deeper research, Joanne McGrath Cohoon and Aspray concluded, even as they reviewed existing studies on the influence of college mentoring and role modeling, interactions with faculty, peer support, images of masculinity in computer departments, pedagogical techniques and curriculum structure, entry barriers disproportionately facing women, and the values by which female college students assess computing and their own abilities.[9]

This intense interest in assessing gender in computing history has not yet fully unpacked a central dimension of this story: the ways that advocates for women in computing, from the 1970s to today, created organizations that aimed to revolutionize thinking about gender and technical work. Their efforts both reflected and promoted the country's growing social, political, and educational concern with increasing diversity and improving the climate for women across science, technology, engineering, and math (STEM) fields more broadly. Conversations about equity became new commonalities in the high-tech world. Public awareness grew, as social media and other popular outlets called attention to episodes of sexism, harassment, and discrimination in computer work. Programs to support girls and women entering non-traditional fields drew high-level support from political leaders, corporate powers, and non-profit groups.

---

7. T. J. Misa. 2010. Defining the problem. In Misa, editor, *Gender Codes*, 3–23, on 7.

8. L. J. Barker and W. Aspray. 2008. The state of research on girls and IT. In J. M. Cohoon and W. Aspray, editors. 2008. *Women and Information Technology: Research on Underrepresentation*, pp. 3–54. MIT Press, Cambridge, MA.

9. J. M. Cohoon and W. Aspray. 2008. A critical review of the research on women's participation in postsecondary computing education. In Cohoon and Aspray, editors, *Women and Information Technology*, 137–180.

It was no coincidence that this period from the 1970s onward, when female presence in computer studies and technical work underwent such rapid growth and then partial collapse, were also the years when women in computing increasingly moved to associate and advocate. This phenomenon happened for four reasons. First, even a temporary or partial rise in the numbers of women in computer fields expanded the base for mobilization, creating a critical mass that could support new advocacy groups. Second, as academic researchers and women in computing themselves discussed the factors limiting women's entry and progress in computer work, such cognizance provided a new spur to combat such obstacles. Third, the growth of women's computer advocacy organizations drew on and gained energy from second- and third-wave feminism, the rise of local, national, and international women's liberation alliances. And finally, the late 20th century and early 21st century witnessed a wider trend in which women across multiple fields of science and engineering felt a drive to organize in hope of addressing their individual and collective concerns.

New national campaigns to promote progress for women in the computer world did not instantly solve issues of inequity, raise female numbers, or resolve workplace gender tensions. Supportive rhetoric was not automatically accompanied by concrete actions to advance women's status in computer work. While leaders' formal speeches, corporate press releases, and advocacy campaigns paid tribute to the advantages of diversifying the nation's technical labor force, women still faced sexual harassment and other incidents meant to undermine their position. Nevertheless, the rise of organized advocacy by and for women in computer fields, along with the rise of public support for diversity in technical work, represent two of the most noteworthy developments in the computer community over recent decades.

So what had shifted over these decades, to bring this outpouring of attention to the gendering of computer knowledge and women's status in computer professions? Archival documents, publications by organizations such as the Association for Computing Machinery (ACM), and media coverage help explain. The answer lies both within the computer world's own history and outside, linked to broader debates over women's place in science, technology, engineering, and math (STEM) and over women's place in modern American society.

In particular, during the late 20th century, women in computing themselves began organizing, together with male allies. Their groups created an institutional base, philosophical justification, and practical opportunities to mobilize support for each other and to promote opening more computer careers to new generations of women. Those women's computing groups in turn evolved, again as a result of shifts both inside the computer world and in wider American life. The campaign to transform the gender dynamics of U.S. technical culture itself became transformed, from the 1960s to today. While it started as small groups of outsiders struggling to win attention, advocacy for women in computing ultimately grew into a high-profile international political, educational and social movement. Initially, simply organizing under the banner "women in computers" made a statement in itself, asserting a hope that collective association could improve the position of female workers in a rapidly evolving field. By the 21st century, advocacy had shifted to reflect a growing understanding that challenges facing women in technical work often involved deeply embedded psychological cues and social messages about gender, making the effort to eliminate bias even more challenging than many observers had initially assumed.

7.2 Gender, Computing, and Organized Advocacy for Women

U.S. science and engineering organizations created in the 19th and early 20th centuries were built by and for men, defined by the masculinized cultures of academia, business, and industry. Individual women, still a rare presence in most technical disciplines, often had to battle to win even grudging acceptance to these national groups. In 1905, the American Society of Civil Engineers (ASCE) offered junior membership to new graduates including Cornell's Nora Stanton Blatch, one of the earliest women in the country to complete an engineering degree. But when Blatch applied for associate membership upon reaching age 32, after her "junior" status expired, ASCE leaders rejected that request and removed her entirely from the organization. She had satisfied requirements for ten years' employment experience, having worked as a drafter, assistant engineer, supervisor, and architect for several New York companies. Blatch filed an unsuccessful lawsuit against the ASCE, which did not admit its first regular female member, Elsie Eaves, until more than ten years later.[10] The equation of merit with masculinity lasted even longer in other organizations. Tau Beta Pi, the only American honor society recognizing excellence across all engineering disciplines, had constitutional rules for seven decades that explicitly limited membership to men. Before 1969, the small but growing number of female students who excelled in college engineering studies could only earn

10. M. W. Rossiter. 1984. *Women Scientists in America: Struggles and Strategies to 1940*. Johns Hopkins University Press, Baltimore; E. T. James, editor. 1971. s.v., "Nora Stanton Blatch Barney." *Notable American Women: A Biographical Dictionary*. Belknap Press of Harvard University Press, Cambridge, MA.

a Tau Beta Pi "women's badge," without voting privileges or other membership rights.[11]

In response, women in science and engineering formed their own organizations, such as the Women's Service Committee of the American Chemical Society (1927), the Society of Women Geographers (1925), and the Society of Women Engineers (1950).[12]

Especially after the mid-1960s, as the ranks of women studying and working in science and engineering expanded (though at starkly varying rates in specific disciplines), more gender-defined STEM organizations appeared, including the Association for Women in Psychology (1969), Association for Women in Mathematics (1971), Women in Cell Biology (1971), and groups within established societies, such as the Committee on the Status of Women in Physics within the American Physical Society (1972).[13] Although each organization differed in the details of its operation, the overarching philosophy was to provide female students and professionals with personal encouragement and practical support to help them persist and succeed often in unfavorable circumstances, to relieve feelings of isolation and frustration in gender-imbalanced settings, and to provide visible proof to command recognition of women's existence in traditionally male fields. During the 1980s and 1990s, still more groups for women in STEM fields arose at the international, national, local and college levels.

Undergraduates, graduate students, and professionals in computer fields participated in those campus or national organizations that brought together women across science and engineering disciplines. Given that pattern of organizing and also given the expansion of computer studies and employment at the end of the twentieth-century, it is no coincidence that those decades also witnessed the establishment of multiple societies for women in computing. Systers, an electronic forum for women's mentoring, networking, and sharing experiences, came

11. H. Petroski. 2011. *An Engineer's Alphabet: Gleanings from the Softer Side of a Profession*. Cambridge University Press, New York; R. Oldenziel. 2003. Multiple-entry visas: Gender and engineering in the US, 1870–1945. In A. Canel, R. Oldenziel, and K. Zachmann, editors. *Crossing Boundaries, Building Bridges: Comparing the History of Women Engineers, 1870s-1990s*, pp. 11–49. Routledge, New York; A. S. Bix. 2013. *Girls Coming to Tech! A History of American Engineering Education for Women*. MIT Press, Cambridge, MA.

12. M. Rossiter. 1995. *Women Scientists in America: Before Affirmative Action, 1940–1972*. Johns Hopkins University Press, Baltimore.

13. M. Rossiter. 2012. *Women Scientists in America: Forging a New World Since 1972*. Johns Hopkins University Press, Baltimore.

together after 13 women began lunch conversations at a 1987 ACM conference. The Computing Research Association's Committee on the Status of Women (CRA-W) appeared in 1991, and Anita Borg's Institute for Women and Technology in 1997. At the grass-roots level, women in computing created formal or informal groups on college campuses or workplaces. In 1977, for example, 6 female computer-science graduate students at the University of California–Berkeley started the support and networking group WICSE, still in existence four decades later. The following year, WICSE hosted "Working in Engineering and Computer Science: A Conference for Women" that attracted an estimated four hundred to eight hundred attendees. WICSE co-founder and conference organizer Barbara Simons (who completed her Berkeley Ph.D. in 1981 and served as president of the ACM from 1998–2000) later recalled, "With the exception of a couple of male deans, all the panelists and speakers were women scientists or engineers. . . . It was a wonderful experience, and there was a lot of comraderie [sic]. Also, I think (hope) that it encouraged a number of younger women to either enter the field, to remain in the field, or to aim for higher goals."[14] Similarly, the University of California at Irvine hosted monthly lunch meetings during the late 1980s, where female faculty could offer women graduate students advice about grant-writing, conference presentations, and other skills needed for career advancement.[15]

A focus here on the Association for Women in Computing (AWC), and the Association for Computing Machinery's Committee on Women in Computing (ACM-W), allows an in-depth comparison of the two groups, illustrating how organized advocacy for women in computers emerged and evolved from the late 1970s to today. AWC's 1970s–1980s approach and ACM-W's post-1991 focus overlapped in many ways. In explaining their motive for existence, both groups started from a philosophy that organized action was essential to improve conditions for women in computers, benefitting technical fields as a whole and opening opportunities more widely to women in future. The AWC aimed "to promote communication among women in computing, to further professional development and advancement of women in computing, and to promote education of women of all ages in computing." Similarly, ACM-W's self-description explained that the group "sup-

14. N. Levelson. 1989. Women in computer science: A report for the NSF CISE cross-disciplinary activities advisory committee. Available at http://sunnyday.mit.edu/papers/nsf.ps (accessed December 30, 2015). See also B. Simons. 2002. Oral history interview by Janet Abbate for the IEEE History Center. Available at http://ethw.org/Oral-History:Barbara_Simons#Getting_her_Ph.D. (accessed December 2015).

15. Levelson, Women in computer science.

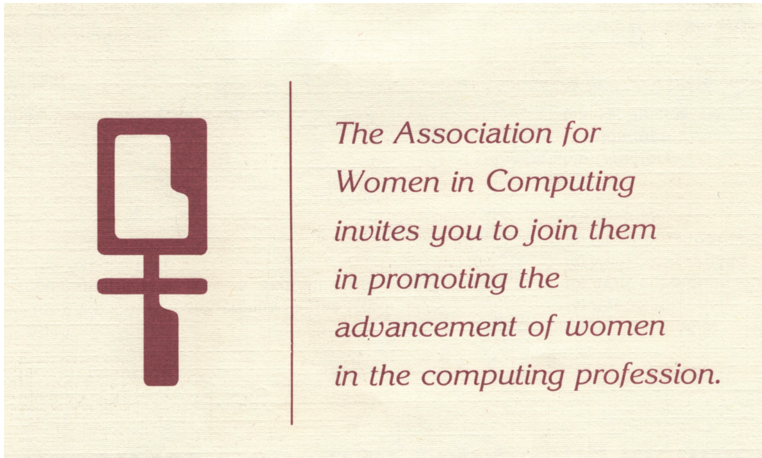

Figure 7.2 **To help define its identity, the Association for Women in Computing created this feminist and tech-savvy logo (1985).** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

ports, celebrates, and advocates internationally for the full engagement of women in all aspects of the computing field, providing a wide range of programs and services to ACM members and working in the larger community to advance the contributions of technical women."[16]

AWC and ACM-W also embraced similar activities and provided parallel types of support for members. For instance, both groups offered awards to recognize promising young women and successful female professionals. Not coincidentally, the AWC's first Ada Lovelace Award, issued in 1983, went to Grace Hopper, who was to later provide such a role model and namesake inspiration for many members of ACM-W. Both groups published newsletters and held conferences to advance networking and a sense of community among women spread across thousands of miles.[17]

16. http://women.acm.org.

17. The figure of Ada Lovelace came to represent a feminist intellectual and political re-claiming of a central role for women in the history of computing. Lovelace became a central figure in steampunk literature and other popular outlets, symbolizing the technologically expert woman. In 2009 British advocates created "Ada Lovelace Day" as a forum to celebrate women in STEM. See R. Hammerman and A. L. Russell, editors. 2015. *Ada's Legacy: Cultures of Computing from the Victorian to the Digital Age*. Association for Computing Machinery and Morgan & Claypool, New York.

More striking, however, are the contrasts between AWC, in its heyday of the late 1970s and early 1980s, and ACM-W, as it grew from the 1990s to today. A comparison between the two groups illustrates key trends in the way that advocacy for women in computers evolved. First, those decades brought an extraordinary scaling-up, not just in membership numbers, but also in the scope of the organization's reach, moving toward international synergy. At the same time, the driving force shifted, moving from an informal enthusiasm sparked by members' personal commitments, toward a more institutionally embedded momentum. In particular, ACM-W was able to leverage its widening connections with ACM and other influential organizations to devote special energy to outreach campaigns aimed at stimulating younger girls' interest in computing. Second, over those decades, early activists' instinctual support for promoting women's place in the computer world was supplemented by informed academic research into issues of gender in the technical world. That shift stemmed in large part from broader conversations inspired by the modern feminist movement and concerns about educational and employment equity, which raised public awareness and political concern about the umbrella topic of women's success in STEM. At the same time, ironically, the spirit of organizing for women in computers shifted away from a wider politically assertive feminism of the 1970s toward a more narrowly focused activism, targeted just at women in the tech world. In particular, advocates embraced popularly oriented efforts to promote computing to younger girls and college women, partnering with powerful institutions and individuals to try reversing the previous decades' downward slide in female participation in computer work.

The AWC was established in 1978; its first planned meeting, at the ACM 1979 Dayton conference, drew over 50 attendees (despite a snowstorm).[18] The AWC held the first annual conference of its own in 1982.[19] In 1983 the AWC had chapters in 12 American cities and by 1985 over 1,000 members.[20] As a new volunteer group, the AWC sometimes struggled in determining where to invest its limited time, energy, and funds. For instance, in 1980, the AWC planned to make a presence at the COMPCON spring meeting, but leaders then worried they had "chosen the wrong conference," since COMPCON would not let AWC hold any sessions or be a good place

18. A. Cochran. March 5, 1979. Association of Women in Computing Records, 1978–1991, Charles Babbage Institute, CBI 49, Box 2, Folder 10.

19. Held in conjunction with the National Computer Conference. AWC press release, May 11, 1982, CBI 49, Box 1, Folder 1.

20. Boston, Chicago, Los Angeles, New York, Pittsburgh, Puget Sound, St. Louis, San Diego, San Francisco, Texas (Lubbock), Twin Cities, and Washington, DC.

for hospitality presence. "COMPCON is not the type of conference that supports or encourages activities such as those proposed by AWC . . . Booths and society sections are *not* in good taste. . . . We voted on [attending] without good data on the conference . . . Politically, it is my understanding that COMPCON is one of those male dominated conference[s] and our presence may have a profound impact."[21]

By contrast, ACM's Committee on the Status of Women in Computer Science grew out of ACM's Committee on Scientific Freedom and Human Rights, derived from a 1991 executive committee formed to study issues of gender in the profession.[22]

From the start, ACM-W enjoyed advantages of alignment with an influential national organization, helping the 21st-century group achieve publicity and clout that the 1970s AWC could never approach. ACM-W went international, hosting meetings in Australia, India, Canada, Cuba, and elsewhere. An ACM-W Europe group, created in 2012, hosted a 2014 womENcourage conference, career fair, and networking focus in Manchester that drew over 200 participants.[23] ACM-W leveraged its position to develop multiple initiatives, including scholarships and funding for student conference attendance, online mentoring, workshops for K-12 teachers on gender and computing, and student chapters around the nation.[24] An official ACM-W project started in 1994, The Ada Project, offered a website clearing house for information distribution, followed by a 2007 Digital Library collection. In 1995, ACM produced the PBS-screened documentary "Minerva's Machine: Women and Computing," which profiled Ada Lovelace, Grace Hopper, Anita Borg, Thelma Estrin, and others, while addressing women's ongoing underrepresentation in technical education and employment. Most notably, by 2004, ACM became a

21. Memo to AWC Executive Board, September 30, 1980, CBI 49, Box 1, Folder 15.

22. This advocacy for women connects to the longer history of ACM's attentions to the social context of technical work. In particular, ACM co-founder Edmund Berkeley persistently but controversially pressed his fellow professionals to engage with the social responsibility of computing (see Chapter 6 in this volume). During the height of anti-Vietnam War fervor in 1972, Berkeley offended Navy Commander Grace Hopper and many colleagues by calling on ACM members to stop all work associated with the military and devote greater effort to applying technical knowledge for public well-being. See B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals*, chapter 7. Association for Computing Machinery and Morgan & Claypool Publishers-ACM Books, New York.

23. V. Hanson, R. Ayfar, and B. Bachmayer. 2014. European women in computing. *Commun. ACM*, 57(7): 5.

24. For an overview, see E. Weyuker. 2009. ACM-W celebrates women in computing. *Commun. ACM*, 52(6): 5.

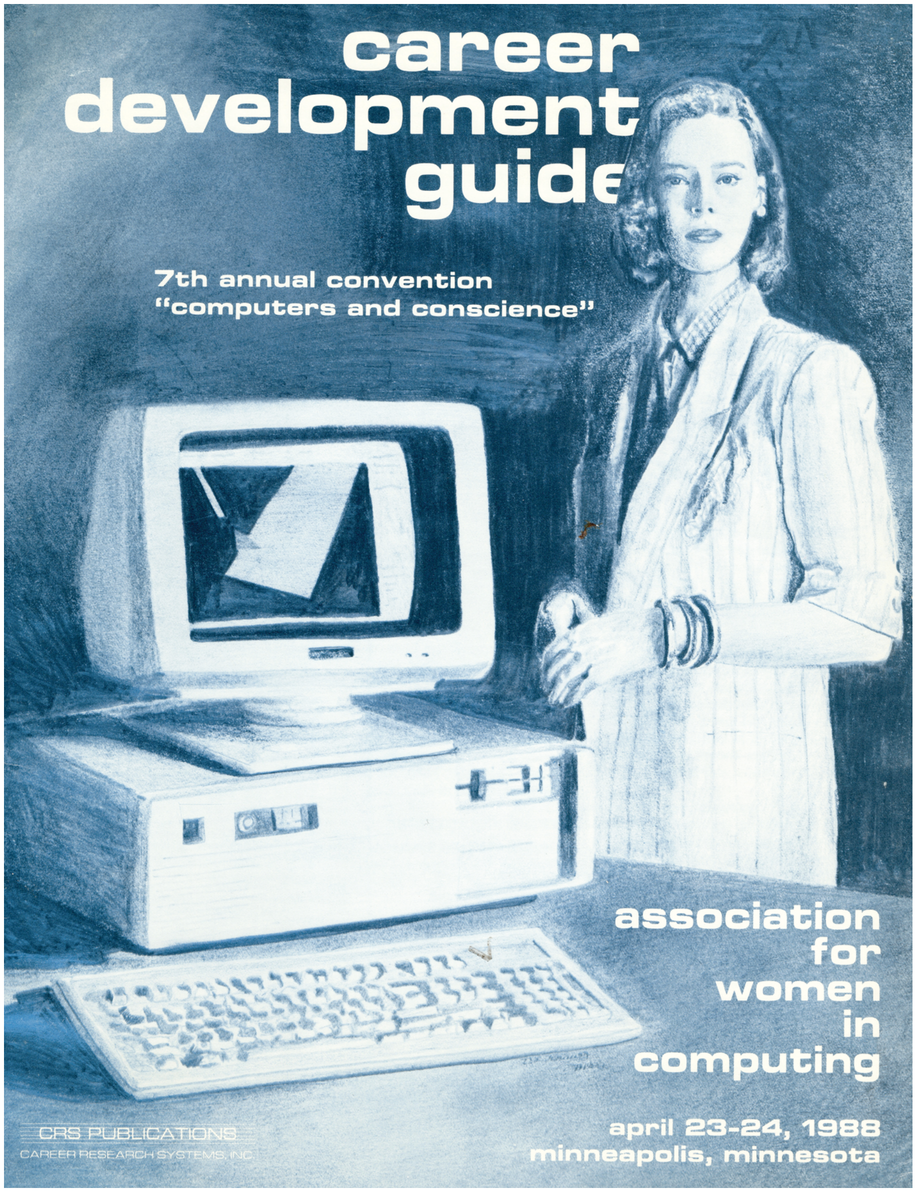

Figure 7.3 **AWC's Career Development Guide (1988) aimed to help women advance in a workplace culture that sometimes seemed unfamiliar and unwelcoming.** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

co-presenter of the Grace Hopper Celebration of Women in Computing Conferences, which originated in 1994; and by 2014, it attracted about 8,000 attendees a year (see Figure 7.4). ACM-W also helped create regional Grace Hopper Celebrations that linked young women with industry role models. ACM-W gained respect and visibility within its parent organization; in 2004 ACM elections, for instance, a number of candidates specifically touted their experience and commitment to work on gender diversity in computing.[25] By 2014, ACM-W had over 20,000 members and almost 90 global chapters.

7.3 Advocacy for Women in Computing and K-12 Outreach

From the 1970s–2000s, the motivation behind advocacy moved away from the AWC's informal enthusiasm sparked by members' personal commitments, toward the ACM-W's more formal and institutional status. This is not to say that 21st-century ACM-W members lacked personal dedication. Quite the opposite: ACM-W supporters often became energized by both their individual questions about working in a male-dominated field, but also by social-media campaigns and well-publicized coverage about ongoing sexism in STEM as a national issue. Support for women in computing evolved from small-scale grassroots campaigns in the 1970s, into a mainstream movement. Issues of STEM diversity commanded attention from the National Academy of Engineering (NAE), National Science Foundation (NSF) ADVANCE programs, the AAUW, Girl Scouts, K-12 educational systems, colleges coast-to-coast, major corporations, and other high-powered allies. Leaders of such institutions framed the goal of broadening diversity in computer fields as a matter of equity, social responsibility, and national competitiveness in high-tech business growth and intellectual innovation. At a 2011 Facebook town-hall, President Barack Obama declared, "[E]very time I come to Silicon Valley . . . I always hear stories about how we can't find enough engineers, we can't find enough computer programmers . . . that's why we're emphasizing teaching girls math and science . . . [W]e want to start making science cool."[26] The White House began hosting annual science fairs in 2010; at the 2015 event, with the theme "Diversity and Inclusion in STEM," the president singled out an Oklahoma Girl Scout

25. Candidate biographies, 2004 ACM Elections.

26. President Barack Obama. Remarks by the President at a Facebook town hall. April 20, 2011. Available at http://www.whitehouse.gov/the-press-office/2011/04/20/remarks-president-facebook-town-hall (accessed December 2015).

team of six-year-old "Supergirls" who built a robot to turn book pages for impaired readers.[27]

ACM and ACM-W both reflected and participated in that 21st-century emergence of high-profile conversations about increasing gender, racial, and other diversity in science and engineering. The two groups poured special effort into K-12 outreach campaigns, often (though not exclusively) targeted to girls. Earlier generations of female computer scientists and male allies had certainly been interested in sparking young women's excitement about technology and confidence to explore possibilities for computer work, but had not always had the resources to mount extensive campaigns. By the 21st century, ACM, ACM-W, and their multiple partners directed local, national, and international outreach programs. In 2009, ACM led a coalition of the NSF, Computing Research Association, Anita Borg Institute, National Center for Women and Information Technology (NCWIT), Google, Intel, and Microsoft, to establish Computer Science Education Week. That celebration—timed annually to honor Grace Hopper's December birthday—aimed to convince policymakers, K–12 teachers, parents, and children themselves to value early exposure to computers. In 2013 ACM expanded those efforts by partnering with Code.org, a nonprofit focused on exposing more female and minority youth to computer science. ACM and ACM-W helped publicize annual "Hour of Code" events that set up coding activities and tutorials so children could create their own drawings, stories, or games linked to pop-culture themes such as Disney characters, Minecraft, Star Wars, and Frozen. Numerous campus ACM chapters organized "Hour of Code" programs in 2014 and 2015, including ACM-W chapters at Kentucky's Bluegrass Community College, Boise State University, and Kansas State.

ACM and ACM-W's girls-outreach efforts expanded in many directions over just a few years. In 2014, ACM joined Microsoft in underwriting a documentary film, "Big Dream," that highlighted seven women pursuing interests in various STEM fields. That same year, ACM-W partnered with Microsoft's MentorNet program, received scholarship funds from Oracle, and claimed a half-million-dollar grant from Google and NCWIT to expand and connect ACM-W student chapter activities. Together with the Computer Science Teachers Association, ACM-W created posters

27. B. Fried. March 23, 2015. The incredible kid-ingenuity on display at the fifth White House science fair. Available at http://www.whitehouse.gov/blog/2015/03/23/incredible-kid-ingenuity-display-fifth-white-house-science-fair (accessed Dec. 2015). NBC News. March 23, 2015. Supergirls conquer Obama at White House science fair. Available at http://www.nbcnews.com/science/science-news/supergirls-conquer-obama-white-house-science-fair-n328661 (accessed December 2015).

and social-media images that showed several scenes of women working with information technology and urged girls, "IT is all about me. Talk to your computer teacher, talk to your school counselor."[28]

Embodying the new 21st-century spotlight on computer advocacy for girls, 2010 brought creation of the ultimate feminized mainstream commercial icon, Computer Engineer Barbie. Girls voted online for Mattel to produce a newsanchor doll. But adults (especially but not exclusively engineers and women) mounted a viral campaign, "Please help . . . Barbie to get her Geek on!" so the firm created both.[29]

Mattel's press release quoted Nora Lin, president of the Society of Women Engineers (SWE), as saying, "[E]ngineer Barbie will show girls that women can turn their ideas into . . . positive impact on people's everyday lives in this exciting and rewarding career."[30] To make Barbie's "Geek Chic" appearance authentic, designers consulted SWE and NAE members,[31] one of whom said, "Barbie will . . . broaden the realm of . . . what feels accessible—being smart, confident, and tech-savvy without sacrificing femininity and fun."[32] Some women engineers found the doll's binary code t-shirt spelling out "Barbie" amusing, but others scorned its trendy leggings as too juvenile, not befitting real female computer experts who demanded to be taken seriously.

28. Computer Science Teachers Association, IT is all about me—Careers in computing. Available at http://csta.acm.org/Resources/sub/CareersFiles/NewCSTA-poster-small-030707.pdf (accessed January 2016).

29. V. Belmont. March 5, 2010. How relevant is computer engineer Barbie? Available at http://www.veronicabelmont.com/2010/03/how-relevant-is-computer-engineer-barbie/ (accessed June 2013); J. Kurth. April 29, 2010. USC professors rally behind Engineer Barbie. *Daily Trojan*. Available at http://dailytrojan.com/2010/04/29/usc-professors-rally-behind-engineer-barbie/ (accessed June 2013); Introducing Computer Engineer Barbie! February 12, 2010. Available at http://www.chipchick.com/2010/02/computer-engineer-barbie.html (accessed June 2, 2013); A. Zimmerman. April 2010. Revenge of the nerds: How Barbie got her geek on. Available at http://online.wsj.com/article/SB10001424052702304198004575171791681002592.html (accessed June 2013).

30. Press release; at B. Barrett. February 12, 2010. Computer Engineer Barbie has a Ph.D. in FUN (and breaking down stereotypes). Available at http://gizmodo.com/5470587/computer-engineer-barbie-has-a-phd-in-fun-and-breaking-down-stereotypes (accessed June 2013).

31. K. T. Bradford. February 16, 2010. How Mattel designed Computer Engineer Barbie to excite adult geeks, young minds. Available at http://blog.laptopmag.com/how-mattel-designed-computer-engineer-barbie-to-excite-adult-geeks-young-minds (accessed June 2013).

32. C. Swaney. September 14, 2010. CMU information networking students to kick off countdown for new Computer Engineer Barbie Doll. Available at http://www.cmu.edu/news/archive/2010/September/sept14_computerengineerbarbie.shtml (accessed June 2013).

Indeed, the whole idea of Computer Engineer Barbie appalled a number of women in tech. One said, "You can't rework a caricature of womanhood with impossibly long legs, pert breasts, and glossy hair into some . . . equal opportunity role-model by giving her a laptop, clothes with circuit board motifs and bright pink (PINK!?!) glasses."[33] Kate Ringland, a University of California informatics Ph.D. student and former ACM-W campus chapter chair, blogged, "There is no way that this Barbie doll has anything to do with the reality of being a computer engineer. I don't know if the doll does more harm than good by portraying such a bizarrely feminine tech freak. . . . I'll admit, we've come a long way from 1992's 'Math is tough' Barbie. . . . [But] I want the daughters of tomorrow to grow up thinking that the best job in the world would be to have the job their moms have—computer scientist."[34]

In ACM-W's summer 2012 newsletter, a column by Claudia Galvan suggested that female computing professionals could "learn . . . [to] move up the career ladder" by paying more attention to Barbie's "soft skills." Rather than simply working hard and waiting for their efforts to win recognition, Galvan suggested, women engineers needed to invest in networking and productive "chitchat," find mentors or become mentors. Barbie's talent for "standing up on your tippy-toes without falling" showed women the importance of gaining confidence to speak up in meetings, "ask for what you want," and "show your accomplishments and ambition." Galvan concluded that professional tech women should also learn from Barbie about "finding your unique personal style," selecting outfits appropriate for a desired promotion and taking "some risks" to make a real first impression, rather than wearing clothing so nondescript that it rendered a woman invisible.[35]

In practice, Computer Engineer Barbie primarily served a symbolic function, underlining the way 21st-century female engineers had staked a claim to more mainstream visibility and acceptance. By positioning "computer engineer" alongside "teacher," "eye doctor," and other "careers" in the "girls' aisle" (itself controversial among opponents of the "pinkification" of girl culture), the doll's existence

33. E. Diffin. February 16, 2010. Does Barbie's new geeky look fit with reality? *BBC News Magazine*. Available at http://news.bbc.co.uk/2/hi/8517097.stm (accessed June 2013). For more discussion, see C. J. Martincic and N. Bhatnagar. 2012. Will Computer Engineer Barbie impact young women's career choices? *Inform. Syst. Educ. J.*, 10(6): 4–14.

34. K. Ringland. January 14, 2012. Nerdy is the new cool. Available at http://kateringland.net/tag/nerd/ (accessed December 2015).

35. C. Galvan. 2012. What can we learn from Computer Engineer Barbie? *ACM-W CIS Newsletter*, 4(1): 21–32.

suggested a validation of aspirations for girls interested in computer technology. For adult women (some of whom kept Computer Engineer Barbie on desks as a personal mascot), the doll's existence represented an amusing identity token.

Unfortunately, Mattel undercut such positive signals with a disastrous next step that illustrated the persistence of gendered stereotypes in technical work. To accompany its new doll, Mattel issued a 2013 book, *I Can Be a Computer Engineer*, that aimed to offer girls a fun account of technical work. The title page showed Barbie bent over two laptops and one desktop screen, bright-pink glasses perched on her nose. The story opened with Barbie telling Skipper that she was "designing a game that shows kids how computers work . . . [to] make a robot puppy do cute tricks by matching up colored blocks." But on the following page, Barbie explained, "I'm only creating the design ideas . . . I'll need Steven's and Brian's help to turn it into a real game!" After a computer virus crashed her own laptop, Barbie failed to reboot it and accidentally infected Skipper's machine by careless use of a flash drive. After her (female) computer-class teacher explained the value of proper security software to Barbie, it is Steven and Brian who recover the lost files, reassuring Barbie that "both laptops will be good as new in no time!" While the book ended with Barbie getting a top grade on her computer game and "happily" announcing, "I guess I can be a computer engineer!" the overall impression suggested that real tech skills remained in *masculine* hands.[36] At a time when more girls than ever were engaged in coding classes, the book never even hinted at Barbie working on programming or engaged in any real engineering beyond the most rudimentary computer use.

Observers savaged the book. One Amazon customer wrote, "As a female computer engineer, I hate this book. Thanks, Mattel, for making it seem like my degree isn't real."[37] Social-media users reposted the book's illustrations with sarcastically rewritten dialogue, linked to the Twitter meme "#feministhackerbarbie." Georgia Tech Ph.D. computing students Miranda Parker (a campus ACM-W officer) and Casey Fiesler created an entire "remixed" approach that repositioned Barbie as handling the coding and Brian as the artistic puppy-creator. Parker's version explicitly included multiple comments about girls' ability to excel in engineering work and

36. S. Marenco. 2013. *Barbie I Can Be . . . a Computer Engineer*. Random House, New York.

37. Amazon.com website for S. Marenco. *Barbie I Can Be . . . a Computer Engineer*. Available at http://www.amazon.com/Actress-Computer-Engineer-Barbie-Pictureback/product-reviews/0449816192/ref=cm_cr_pr_paging_btm_next_18?ie=UTF8&showViewpoints=1&sortBy=helpful&pageNumber=18 (accessed December 2014). See also P. Ribon. November 17, 2014. Barbie fucks it up again. Available at http://pamie.com/2014/11/barbie-fucks-it-up-again/ (accessed December 2014).

ended with Barbie telling her high-school computer teacher, "I learned a lot more about [P]ython and I'm going to teach myself Java next. . . . I hope classes like yours encourage more girls to go into STEM fields."[38]

7.4 The Shifting Nature of Advocacy for Women in Computers

The 21st-century growth of American public and professional attention to issues of female underrepresentation in computer work was no accident. Rather, it stemmed from broader concerns about increasing diversity in STEM fields, which in turn drew on and connected with a rising sense of the social, psychological, and political complexity of gender. Parents, teachers, educators, policymakers, and both female and male scientists and engineers themselves dove deeper to assess what factors perpetuated female underrepresentation in STEM work, and such analysis in turn informed and altered advocacy.

As new organizations for women in STEM emerged during the late 1960s and early 1970s, their leaders developed programs based on their immediate impressions of what activities could prove most beneficial in encouraging women to enter and persist in non-traditional specialties. When assessing conditions facing women in computer work, AWC members remembered their own experiences with teachers who dismissed girls' math ability. They had first-hand knowledge of how infuriating it was to have colleagues who refused to take professional women seriously, shouted them down at meetings, harassed them, offered patronizing help, or assumed women were less-qualified, affirmative action hires. Drawing on such personal feelings, AWC members focused on setting up mentoring and networking systems as a measure that could help not just them, but also women newly entering computer work. The organization aimed to create "a new girl network," that they thought might one day match "the old boy network."[39] More than that, AWC hoped to "project a strong, positive image of women in the computing industry and in the workplace in general." Advocates suggested that as individual women attained professional stature and success, those small triumphs ultimately would redound to create more positive gender conditions across the industry.

---

38. "The Internet's Best Responses to Computer Engineer Barbie," http://gizmodo.com/the-internets-best-responses-to-computer-engineer-barbi-1660660376 (accessed December 2014). C. Fiesler. Barbie I can be a computer engineer: The remix! Now with less sexism! Available at http://cfiesler.files.wordpress.com/2014/11/barbieremixed.pdf (accessed December 2014).

39. *Computerworld* (June 14, 1982): 20; and AWC Publications Working Paper, August 8, 1981, CBI 49, Box 1, Folder 17.

Three decades later, ACM-W had joined other women's organizations in promoting similar networking and mentoring activities, newly informed by the wealth of academic and advocacy literature investigating gender issues in STEM. Research by psychologists, sociologists, educators, scientists, and engineers emphasized that there was no single reason (and thus no simple solution) why many women remained reluctant to embrace or persist in male-identified fields such as computers. The problem was deeply entangled with persistent gendered assumptions about femininity/masculinity, stereotypes about hackers and nerds, discrimination and harassment in the workplace, and psychological barriers such as "stereotype threat" and the "imposter problem."[40]

ACM and ACM-W themselves contributed to this new interest in analyzing the gender imbalance in scientific and technical disciplines. Notably, in 1990, *Communications of the ACM* documented women's underrepresentation in both undergraduate and graduate-level computer degree programs and in employment. ACM readers learned that in 1987–88, female employees held just thirty percent of computer-science jobs, and 6.5 percent of faculty positions in the top 158 Ph.D.-granting universities as of 1988–89. In 1983–86 (peak years for women gaining undergraduate CS degrees), only ten to twelve percent of doctoral degrees in computer science went to female graduates.[41] Offering thoughts about possible roots of this "pipeline shrinkage," ACM's Committee on the Status of Women in Computer Science cited aspects of the tech world that created a "chilly" climate for women, such as boy-friendly gaming culture, the masculinized hacker mentality, and

40. For overviews, see J. Margolis and A. Fisher. 2003. *Unlocking the Clubhouse: Women in Computing*. MIT Press, Cambridge, MA; D. E. Shalala, chair. 2007. The Committee on Maximizing the Potential of Women in Academic Science and Engineering. *Beyond Bias and Barriers: Fulfilling the Potential of Women in Academic Science and Engineering*. National Academies Press, Washington, DC; S. J. Ceci and W. M. Williams, editors. 2007. *Why Aren't More Women in Science?* American Psychological Association, Washington, DC); M. Wyer, et al., editors. 2013. *Women, Science, and Technology: A Reader in Feminist Science Studies*. Routledge, New York; J. S. McIlwee and J. G. Robinson. 1992. *Women in Engineering: Gender, Power, and Workplace Culture*. SUNY Press, Albany; C. Hill, C. Corbett, and A. St. Rose. 2010. *Why So Few?: Women in Science, Technology, Engineering, and Mathematics*. AAUW, Washington DC; and C. Corbett and C. Hill. 2015. *Solving the Equation: The Variables for Women's Success in Engineering and Computing*. AAUW, Washington DC.

41. A. Pearl, et al. 1990. Becoming a computer scientist: A report by the ACM committee on the status of women in computing science. *Commun. ACM*, 33(11): 47–57. See also K. Frenkel. 1990. Women and computing. *Commun. ACM*, 33(11): 34–46, which among other things, provided ACM readers with a substantive bibliography on gender and computing, plus a report on a workshop, "In Search of Gender-Free Paradigms for Computer Science Education," held at the National Educational Computing Conference in June 1990.

ongoing issues of discrimination. That report also linked women's underrepresentation in computer studies and employment to wider factors, such as college women's fears about their physical security on campus, gendered self-esteem issues, and the perpetual problems of juggling work and family that employees both inside and outside the tech world faced. ACM's Committee on the Status of Women offered seventeen suggestions for ways that computer-studies programs and employers might help to redress the imbalance. The report recommended that institutions create workforce re-entry programs for women, effective grievance procedures, and parental leave policies.[42]

Diversity advocates inside ACM extended the self-assessment over subsequent years. In 1992, an ACM-funded Committee to Assess the Scope and Direction of Computer Science and Technology echoed calls for "outreach" to counter the "unfortunate" underrepresentation of women and minorities.[43] A 1997 article by ACM-W member Tracy Camp sounded the alarm that "the percentage of bachelor's degrees awarded in CS to women decreased almost every year over the last decade."[44] ACM's constituency did not always welcome such attention to gender; one reader called the report "junk science," asking, "What problem is the ACM Committee on Women in Computing trying to solve? Where is the evidence we do not have equal opportunity? Should I not use electricity unless 50 percent of the power plant operators are women? Who is the Internet Gestapo that is keeping women from participating?"[45] ACM-W's Tracy Camp responded, "The forces at play are not 'Gestapo' or conspiracies. They are much more subtle and complex."[46] Camp's comment signaled growing attention to what became known as "micro-discrimination," small but cumulative challenges that added up to discourage women and minorities.

ACM conferences and publications during the 1990s and 2000s featured increasingly visible articles, papers, and panels analyzing gendered experiences and issues in computer education and work, research often created or promoted by ACM-W. A small sampling includes reports on a backlash against women linked to affirmative-action resentment,[47] departmental variation in female retention rates,[48] computer

42. Pearl, et al. Becoming a computer scientist.

43. J. Hartmanis. 1992. Computing the future. *Commun. ACM*, 35(11): 31–40.

44. T. Camp. 1997. The incredible shrinking pipeline. *Commun. ACM*, 40(10): 103–110.

45. E. F. Glynn. 1998. More studies needed. *Commun. ACM*, 41(3): 27.

46. T. Camp. 1998. Response. *Commun. ACM*, 41(3): 27–28.

47. L. M. Zurk, et. al. 1995. When an advantage is not an advantage. *Commun. ACM*, 38(12): 17–18.

48. J. M. Cohoon. 1999. Departmental differences can point the way to improving female retention in computer science. *Technical Symposium on Computer Science Education*, pp. 198–202. ACM.

games designed to engage girls,[49] female robotics-programming classes,[50] gendered collaboration in CS classrooms,[51] and defensive communication behavior.[52] In 1995, 2002, 2004, and 2011, various ACM publications devoted special clusters or entire issues to analysis of gender, computing, and the goal of broadening participation by underrepresented groups.

Through the 1990s and after, ACM-W's discussion of gender and computer work connected with a broader feminist advocacy on issues such as ongoing gender biases in K-12 education. In 1992, for instance, the influential report *How Schools Shortchange Girls*, issued by the American Association of University Women (AAUW), contended that in elementary schools, boys received disproportionate amounts of attention from teachers and that curriculum content also neglected girls.[53] In 2010, the AAUW produced a full report, funded by the NSF, addressing the question of why women's "progress" in science and engineering had been significantly "slower" than in other "historically male fields such as business, law, and medicine." *Why So Few* focused on the way "social and environmental factors contribute to the under-representation of women in science and engineering." AAUW researchers emphasized the impact of stubborn stereotypes about girls' mathematical limitations, the danger of girls' self-doubting their abilities, and the harms of both workplace bias and "implicit bias" in STEM.[54]

Singling out computing and engineering as having the most entrenched "masculine culture," a follow-up 2015 AAUW report explained, "[W]omen remain less well represented in engineering than in any other STEM field, and computing has the dubious distinction of being the only STEM field in which women's representation has steadily declined throughout the past few decades." *Solving the Equation* elaborated on the principle that "women often feel as if they don't fit or belong in these fields," due to "structural and cultural barriers" such as the rigidity of

49. C. M. Gorriz and C. Medina. 2000. Engaging girls with computers through software games. *Commun. ACM*, 43(1): 42–49.

50. M. Schep and N. McNulty. 2002. Experiences with using robots in an all-female programming class. ITISCE '02.

51. S. Berenson, et al. March 2004. Voices of women in a software engineering course: Reflections on collaboration. *ACM J. Educ. Resourc. Comput.*, 4(1).

52. K. Garvin-Doxas and L. J. Barker. March 2004. Communication in computer science classrooms: Understanding defensive climates as a means of creating supportive behaviors. *ACM J. Educ. Resourc. Comput.*, 4(1): 1–18.

53. The American Association of University Women. 1992. *How Schools Shortchange Girls: A Study of Major Findings on Girls and Education*. AAUW Educational Foundation, Washington, DC.

54. C. Hill, et al., *Why So Few?*, pp. ix, xiv.

Figure 7.4 **The Grace Hopper Celebration of Women in Computing (2010) exemplifies 21st century female STEM advocacy as a high-power multicultural movement.** (Courtesy of Anita Borg Institute © 2016)

the technical curriculum, continued exclusion from male social networks when job hunting, lack of mentors, poor access to training and development, feelings of isolation, work-family conflicts, and intimidating work environments. The report emphasized ongoing studies that documented the power and therefore the danger of stereotypes and stereotype threat, microinequities and microaggressions, favoritism and biased evaluations, and the stubborn association of STEM with masculinity. "Evidence shows that bias against women, particularly in stereotypically male domains, is widespread. . . . [W]e are all influenced by gender biases, whether or not we consciously endorse them."

The AAUW praised Harvey Mudd College for taking its computer science program from an average of twelve percent female majors to forty percent over five years, by, among other things, offering female first-year students free trips to the annual Grace Hopper Celebration (GHC). "Results indicate that attending GHC . . . is effective in addressing the barriers that keep women from choosing to study computer science. . . . One of four women who came to Harvey Mudd not considering a computer science major and who attended GHC ended up majoring in computer science." The report recommended that other colleges work to duplicate Harvey Mudd's success by bringing undergraduates to women's computer conferences, especially the smaller and thus more accessible and less overwhelming regional Grace Hopper Celebrations. "Taking even a few students can change their mindset and have an important effect on a school's program."[55]

21st-century advocacy for women in computers thus came to both stimulate and reflect extensive research about gender and STEM that simply had not yet existed in the 1970s. Similarly, comparing early organizing of the AWC to later activities of the ACM-W reveals that advocacy underwent a notable shift in emphasis over those four decades. During the late 1970s, early AWC exchanges and activities displayed a politically assertive, wide-ranging feminism. Several decades later, ACM-W's direction appeared distinctly different, targeted more exclusively to issues directly linked to women in computing.

The AWC's creation in 1978 came not coincidentally at a time of key transitions in three areas. First, the field of computer work had gained visibility outside the worlds of higher education and high-tech corporations, catching the eye of a more general American audience. In 1981, for example, *Women's Day* consulted and quoted AWC's president for an article promoting computer careers for women.[56] Second, the 1970s brought expanded discussion of women's access to traditionally male positions, in politics, professions such as law and medicine, and blue-collar fields such as construction. Third, the ongoing second-wave feminist movement directed ongoing attention and activism to fighting injustices in education, employment, and social opportunity.

Many in the AWC regarded feminism, professionalization, and technology as entwined issues, in an era when women's careers still remained controversial in many quarters. The AWC promoted feminist-influenced thinking about women in the family and workplace, sponsored discussions about unequal pay, and openly addressed other aspects of gender inequity. Group members worked to call attention to discriminatory assumptions embedded in both the technical world and American society more broadly. For instance, one member informed the AWC that she

55. C. Corbett and C. Hill. 2105. *Solving the Equation: The Variables for Women's Success in Engineering and Computing*, pp. 8, 34, 48, 53, 80–81, 83. AAUW, Washington DC.

56. Where the money is: Computer careers. *Women's Day*, March 10, 1981: 53–58, 102.

had complained to the authors of a PASCAL textbook about their sample problem that described a database automatically changing a wife's name to her husband's upon marriage.[57]

Most notably, one of AWC's earliest actions, in 1980, was to pass a resolution declaring, "Whereas, women continue to be one of the most discriminated against and exploited groups in the nation, witness the fact that they earn an average of only 3/5 of what men earn . . . the [AWC] endorses the Equal Rights Amendment . . . [and] intends to hold national meetings only in those states that have ratified the [ERA]." AWC leaders acknowledged that since AWC had no conferences of its own at that time, it had no economic power to make a boycott meaningful and no direct power to influence where other groups held conferences at which AWC might appear. Nevertheless, taking the position seemed important to some; as one member earlier wrote, "I am glad to finally see an organization of technical women with balls enough to take a firm stand on boycotting states that have not passed the ERA."[58] But the step "caused alot [sic] of dissention within AWC," and instead of boycotting Chicago's 1981 National Computer Conference, AWC chose to appear with black armbands and pro-ERA buttons.[59] Some AWC members feared being seen as "just another bra-burning, radical organization." There was worry that if AWC became seen as too "political," it might scare off potential members and alienate colleagues, especially in the conservative mood of the Reagan era. In 1985, at the height of what came to be known as the "backlash" against feminism, AWC president Virginia Walker wrote, "I recently read an article [about] . . . battles raging in the workplace which [we] thought had been won five or ten years ago. Somehow it is hard to consider that all of the progress which has been made for women in the workplace is eroding as we become complacent about our successes. We need to find ways to bring this to the attention of the women in our profession, as we need to find ways to stem the tide back to the oppressive past."[60]

During the 1980s, such caution led to at least a partial retreat from overtly feminist politics in the AWC. Organization activities instead often focused on helping

---

57. AWC forum. *AWC Newsletter*, January 1980.

58. AWC. Resolution on the Equal Rights Amendment. May 18, 1980. CBI 49, Box 1, Folder 44. Alice Jackson to Nancy Mae Bonney, October 14, 1980, CBI 49, Box 1, Folder 15. Shirley McCarty to Nancy Bonnie Bryan, November 20, 1979, CBI 49, Box 1, Folder 44.

59. See *Computerworld* (March 23, 1981): 48.

60. L. Sauchin. August 31, 1985. CBI 49, Box 1, Folder 24; and R. Gordon. AWC—More than 'Good Old Girls' club. *Computer Careers News*; Virginia Walker to AWC Board of Directors, August 23, 1985, CBI 49, Box 1, Folder 1.

individual women climb the employment ladder. Chapter programs, newsletter articles, and members' speeches instructed members on the secrets of mastering "corporate know-how" and the "games mother never taught you." Other events promised to help women navigate the politics of management and promotion, as well as learning about fringe benefits, investing, the politics of office dress and effective image presentation, public speaking, two-career couples, and more. Thelma Estrin described such topics as "designing and debugging careers." Such issues occasionally reasserted connections to broader feminist commentary. For example, a 1980–1981 series of Washington DC AWC chapter meetings focused on "Power . . . a topic like sex or religion: not discussed in polite company. . . . Let's get the taboos out of the closet."[61]

By the 21st-century, mainstream feminist-oriented organizations had moved to join in advocacy for women in computers and in STEM more generally, through both rhetoric and activity. Yet at the same time, the movement for women within the computer community often embraced a more narrow, internalist perspective. At the 2014 Grace Hopper Celebration (GHC), Microsoft CEO Satya Nadella made headlines by declaring that the computer world was a meritocracy that naturally rewarded superior performance and thus "women who don't ask for raises" should trust in "good karma" for eventual compensation.[62]

The ensuing uproar gave advocates for women in computing a vehicle to draw national attention to gender inequities in Silicon Valley. Valerie Barr, ACM-W's chair, editorialized, "The situation provided a great opportunity for people to talk about the fact that meritocracy does not work when there is implicit bias. Nadella may think that the Microsoft review process is fair and unbiased, but given that women in tech in the U.S. earn only $0.86 for every $1.00 earned by men, he should seriously research what the actual numbers are for Microsoft, and then adjust salaries accordingly." Barr continued by advising young women attending the GHC not to let recruiters offering "fancy swag" seduce them into believing that "everything is great now in the tech world. . . . Nadella's comments serve as a reminder that women entering the field still have to be prepared to advocate for themselves when they negotiate starting salaries and subsequent raises."[63] Nadella apologized within hours and clarified his remarks with the message,

61. Handout, the Capital area chapter of AWC; n.d., ca. Nov. 1980, CBI 49, Box 3, Folder 11.

62. C. Riley. October 10, 2014. Microsoft CEO to women: Not asking for a raise is 'Good Karma'. Available at http://money.cnn.com/2014/10/09/technology/microsoft-ceo/ (accessed January 2016).

63. V. Barr. October, 2014. Welcome from the ACM-W chair. *ACM-W Connections*, p. 1.

"I wholeheartedly support programs at Microsoft and in the industry that bring more women into technology and close the pay gap. I believe men and women should get equal pay for equal work . . . if you think you deserve a raise, you should just ask."[64]

ACM and ACM-W capitalized on the high-profile media coverage to extend dialogue about the gendered complexities of technical engagement. The very next month, ACM's Special Interest Group on Computers and Society published a special newsletter issue on "action items we can take to address the gender gap in computing." Purdue's Gene Spafford (ACM Fellow, Council member, and awardee) deplored poor treatment and marginalization of women in cybersecurity work, advising misbehaving or clueless men on how to treat women as professional equals and advising women to engage with ACM-W, NCWIT, CRA-W, and similar groups offering valuable support. Jane Stout and Tracy Camp reviewed social-science theories explaining the lack of diversity in computing and recommended more mentoring for both K-12 students and more senior women, creating more inclusive environments for computer work, and auditing company hiring and promotion practices to root out bias. The newsletter's final article linked the gender imbalance in hardcore gaming culture and the misogynist cyber-mobbing of female critics to a broader crisis of "everyday sexism." Authors Michael James Heron, Pauline Belford, and Ayse Goker carefully linked "the toxicity in online popular culture" to the "heavily sexualized abuse" of women in the Scottish independence debate and the "#yesallwomen" Twitter hashtag that let women voice their frustrations with harassment and discrimination.[65] Their analysis offered a distinct contrast to many other pieces by women's computer advocates and in the popular press that treated gender issues regarding computers as divorced from gender issues in the rest of K-12 education, college experiences, and working conditions.

Discussion of the Nadella episode spurred attention to the gender-skewed job numbers and earnings gap at large firms such as Microsoft, but rarely connected to a more systematic analysis of the national male-female pay differential. Commentary on women's attrition rates in computer studies typically failed to assess

64. M. Clinch. October 10, 2014. Microsoft CEO apologizes for gender pay gap comments. Available at http://www.cnbc.com/2014/10/10/microsoft-ceo-apologizes-for-gender-pay-gap-comments.html (accessed January 2016).

65. D. Weikle. 2014. Letter from the editor. *ACM SIGCAS Comput. Soc.*, 44(4): 3; J. Stout and T. Camp. Now what? Action items from social science research to bridge the gender gap in computing research, ibid., 5–8; G. Spafford, We are out of balance. ibid., 9–12; M. J. Heron, P. Belford, and A. Goker. Sexism in the circuitry: Female participation in male-dominated popular computer culture, ibid., 18–29.

lingering gender bias in the overall structure and climate of higher education. Notes about women's attrition rates and negative experiences in computer employment usually treated Silicon Valley as a world onto itself, without engaging the bigger political, legal, economic, and everyday challenges facing female workers other than tech professionals. Advocacy for women and diversity in computers had become more mainstream but also more internalist and politically "safe," less directly involved with any wider 21st-century feminist activism.

7.5 Conclusion

Over five decades, many factors that drove female students, professors, and professionals, along with male allies, to devote precious time to AWC, ACM-W, or similar organizations remained constant. In particular, advocates repeatedly expressed frustration that gender equity in the computer world remained elusive, alternating with the hope of making a difference for even a few young women. Colorado software engineer Darlene Banta wrote in a 2013 ACM-W newsletter that before attending a regional Celebration of Women in Computing, she "had forgotten what it was like to interact with other women in computing. . . . the comfort level I experienced when interacting with others technically and casually is the true value of women-focused conferences. The little doubtful voice in the back of my head vanished! . . . knowing that other women share my experiences, knowing that this is the right career for me."[66] That sense of community remained vital to discussions and activism for gender and computing from the 1970s into the 2010s.

In comparison to AWC's efforts of the 1970s and 1980s, ACM-W's work of the 1990s and 21st-century grew international and developed widespread new initiatives that brought unprecedented visibility and resources to programs for mentoring, support, and education. Yet even as ACM-W attained new levels of financing, prominence, and mainstream endorsements, its members and researchers also concluded that the task of promoting female success in computing remained a deeper challenge than many advocates had earlier recognized, for complex psychological, educational, and social reasons. By 2015, the numbers on female participation in computer studies, employment, and professionalization had not suddenly jumped upward; the outpouring of practical, financial, and rhetorical support for diversification did not serve as a magic spell that instantly swept more girls and women into stereotypically masculine fields.

66. D. Banta. 2013. Inspiration lost and found. *ACM-W CIS Newslett.*, 4(2): 22–24.

Women's conversations about the computer field still continued to revolve around familiar themes—the harmful influence of male-geek stereotypes, challenges of balancing family and career, and the need for greater outreach to K-12 girls to capture their technical curiosity before self-questioning and social pressure frightened them away from computers. But computer culture had unquestionably changed over five decades, as corporations, government, and academia extended new commitments to demonstrate public support for diversity in computing. 1970s women in U.S. computing often expressed a sense of loneliness; while 21st-century women were well aware of ongoing underrepresentation, students and young professionals who chose to seek community could find more resources than ever to address multi-faceted questions about gender, professionalization, and technical knowledge.

8

The Development of Computer Professionalization in Canada

Scott Campbell[1]

In this chapter I explore the leading Canadian computing organizations and professionalization movements from the 1950s–1970s. In particular, my goal is to review the challenges faced in Canada by those who sought the benefits of professionalization or just the camaraderie of a social organization. Technologically, the development of the computer industry in Canada was similar to that in the U.S., with some differences relating to the relative scale of the countries and the lack of a general-purpose electronic computer industry. While many Canadians in the industry felt that the differences were significant, the ultimate challenges of professional organization were substantially similar on both sides of the border. Among the common questions: Who ought to be considered a computing professional? How were educational standards to be maintained? Who should be in charge of accreditation? Why bother to professionalize at all? As a general rule, Canadians

1. I'd like to thank the Charles Babbage Institute staff, particularly Arvid Nelsen, for help with ACM, DPMA, IFIP, and other related archival material. Chuck House, Jim Cortada, and Bill Aspray provided helpful feedback to an earlier draft of this chapter presented at the Society for the History of Technology in Albuquerque, NM in 2015; Tom Misa organized the excellent ACM history panels. My research assistant, Wendy Stocker, offered invaluable assistance at various stages of research and writing. I'd also like to acknowledge financial assistance from the Charles Babbage Institute's Arthur Norberg Travel Fund and the ACM History Fellowship.

faced these problems a few years after their Americans neighbors, but eventually adopted very similar answers, with some occasional reluctance.

After briefly describing the differences between the two countries and their computing capacities, I will discuss the history of the principal computing organizations that operated in Canada from the early 1950s to the early 1970s, and compare their approaches to professionalization. The Association for Computing Machinery (ACM) provides the entry point to this story as the first computing organization to establish a foothold in Canada in 1952. However, its early presence was overshadowed by the arrival of the Data Processing Management Association (DPMA) the following year, and which likely achieved greater success due to the different computing context in Canada. Finally, I will tentatively analyze the roles of Canadian identity and technological nationalism, and how these relate to the problems of professionalization. The battle over accreditation and professionalization that took place between the ACM and DPMA in the 1960s in the U.S. was repeated in Canada during the same period and wrapped more tightly in a national flag.

8.1 Rise of Canadian Computing

Electronic computing in Canada between the early 1950s and the early 1970s can be broadly characterized as people and organizations learning how to do data processing and scientific computing effectively. Unlike developments in the U.S. or the United Kingdom, the Canadian computing industry developed with almost no general-purpose computer hardware research, development, or commercialization. This is not to say that no electronic computing hardware was designed in Canada during this period, but rather that virtually every electronic computer used in banks, businesses, universities, government offices, and manufacturers was imported. One notable exception is the FP-6000, a general purpose, commercial mainframe designed at Ferranti-Packard Canada in the early 1960s. It sold poorly in North America and in 1963 Ferranti UK, the parent corporation, sold all its non-military computing assets—including the FP-6000—to ICT, which turned the design into a spectacular success in Europe.[2]

The first three computers installed in Canada between 1952 and 1955 were intended for scientific computing, but in the second half of the decade, electronic

2. J. N. Vardalas. 2001. *The Computer Revolution in Canada*. MIT Press, Cambridge, MA. By the mid-1970s, microprocessors and personal computing helped open opportunities in Canada for new entrants, such as the MCM/70. See Z. Stachniak. 2011. *Inventing the PC*. McGill-Queen's University Press, Montreal.

data processing for business and government came to dominate; this is a pattern that was common in many other countries. By 1958 there were about 50 computers installed or on order in Canada, and most were for use in offices and data processing departments.[3] A 1962 survey by the federal government located over 300 computers in the country.[4] Of course, these numbers pale in comparison with those for the U.S., the primary point of comparison, which had more than 13,000 computers in 1962 and more than 20,000 in 1964.[5]

Such comparisons provoked some fears among Canadians that they were underusing computers or, worse, falling behind. A 1964 government report offered more realistic insights into the relatively low number of computers to be found in Canada. The country was home to fewer large firms that could afford a computer, and Canadian clerical salaries were lower than those in the U.S., making a shift to electronic data processing less attractive in Canada. A tariff on computer imports also made computers more expensive in Canada. Many American companies with Canadian subsidiaries, which might have purchased or rented a computer, "tend[ed] to centralize their computing facilities. . . . [at] head offices in the United States."[6] Finally, the report alluded to a particular Canadian character trait—resistance to change and new technology—that might also have slowed growth.

Ultimately, despite any quantitative differences, the patterns of use were quite similar. By the mid-1960s, a majority of computers in both countries were medium-scale, solid state commercial systems intended for data processing. To use such technology effectively, both countries needed a body of experienced managers and skilled workers. Like their U.S. counterparts, many Canadian computing professionals working in accounting, payroll, or other financial departments had adopted the new technology. A smaller number had closer ties to science, math, or engineering, and what would eventually come to be known as computer science.[7]

3. Z. Stachniak and S. Campbell. 2009. *Computing in Canada: Building a Digital Future*. Canada Science and Technology Museum, Ottawa.

4. Department of Labour. 1962. *A Second Survey of Electronic Data Processing in Canada*. Ottawa, Canada.

5. J. C. McDonald. 1964. Impact and implications of office automation. Occasional Paper 2. Economics and Research Branch, Department of Labour, Ottawa.

6. Ibid., 9.

7. See T. Haigh. 2001. The chromium-plated tabulator: Institutionalizing an electronic revolution, 1954–1958. *IEEE Ann. Hist. Comput.*, 23(4): 75–104; and N. L. Ensmenger. 2012. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*. MIT Press, Cambridge, MA.

8.2 ACM's Early Start in Canada

Over a dozen computing organizations, user groups, and societies would eventually appear in Canada, beginning in the early 1950s. They can be divided into three broad categories. Some, like the Association for Computing Machinery (ACM) and the Data Processing Management Association (DPMA), were based in the U.S. with clear American perspectives and interests but permitted local Canadian chapters. Others, such as the International Federation for Information Processing (IFIP), explicitly offered an international forum and an organizational structure that acted as an umbrella for the many national computing societies around the world. The third group, which included the Computing and Data Processing Society of Canada (CDPSC)—later known as the Canadian Information Processing Society or CIPS—had an exclusively Canadian agenda and membership.[8]

As with the hardware itself, the first computing organization in Canada came from outside the country. The Association for Computing Machinery held its seventh national meeting at the University of Toronto in September 1952. This was the second national ACM meeting of the year, the first ever outside the U.S., and the first computing conference in Canada.[9] Like many of the early ACM meetings, it was held on a university campus and resulted from a local initiative. In 1951, Calvin C. "Kelly" Gotlieb, the day-to-day director of the school's computing center, invited the ACM to Toronto and then served as chair of the local organizing and program committees.

The University of Toronto's Computation Centre had been created in 1948 in response to news of the rapid development of computing technology in the U.S. A mathematical group within the Centre acted as a scientific computing service to Canadian scientists, and an electronics group of faculty and graduate students worked on a prototype computer influenced by von Neumann's well-known design at Princeton's Institute for Advanced Study. By 1951, the prototype worked—poorly—but they never finished a full-scale version. The first *useful* computer was a commercially manufactured Ferranti Mark I, imported from Manchester, UK in April 1952. Installation and testing took the entire summer, but it was partially operational in time for the ACM delegates in September.[10]

8. A fourth category—user groups, such as SHARE—will not be discussed, because the machine-centric perspective of a user-group tends to exclude national viewpoints that will be explored below.

9. F. L. Alt. 1962. Fifteen years of ACM. *Commun. ACM*, 5(6): 300–307.

10. S. M. Campbell. 2006. The premise of computer science: Establishing modern computing at the University of Toronto (1945–1964). Ph.D. thesis, Institute for the History and Philosophy of Science and Technology, University of Toronto.

The "international" setting and nearly assembled Ferranti computer likely helped to attract attendees from the United Kingdom, including Maurice Wilkes, director of the EDSAC project at Cambridge, and several people associated with the Manchester-based Ferranti computer. Notably, Christopher Strachey gave a talk on non-mathematical programming and his famous checkers program. Toronto's computer was working well enough for him to display the game board, but not to play a game.[11]

Through these visitors, the ACM meeting acted as a nexus of technological transfer, in passing programming knowledge about the Ferranti Mark I from the UK to Canada. Although the Computation Centre had been building an electronic computer since 1948, little time had been spent considering the problems of writing programs. Only two staff members had much in the way of experience: Kelly Gotlieb had traveled to Manchester earlier that year for a few weeks of training and Beatrice Worsley had spent several years at Cambridge completing her Ph.D., in the process learning how to use both EDSAC and the Manchester machine.[12] Fortunately, D.G. Prinz, employed by Ferranti in its programming group, was attending the ACM meeting, and spent time tutoring the Computation Centre staff. He retyped his programming manual from memory and explained various aspects of the Mark I.[13] Strachey contributed to the lessons, and stayed after the meeting to assist the Computation Centre staff during the first few months of computer operations. He also directed the development of its first large program, used for backwater calculations of the St. Lawrence Seaway and Power Project, and which taught everyone involved a great deal about hardware and software reliability.[14]

Of course, a majority of ACM attendees and presenters were from the U.S. They represented major projects from Princeton; the National Bureau of Standards; the Los Alamos, Oak Ridge, and Argonne national laboratories; and the Ballistics Research Laboratory and its famous computing facilities, which included the ENIAC, EDVAC, and some Bell Relay calculators.[15] Undoubtedly, many Americans would have been curious about the Ferranti Mark I, an unfamiliar, British-built machine.

11. M. Campbell-Kelly. 1985. Christopher Strachey, 1916–1975: A biographical note. *Ann. Hist. Comput.*, 7(1): 19–42, on 26.

12. S. Campbell. 2003. Beatrice Helen Worsley: Canada's female computer pioneer. *IEEE Ann. Hist. Comput.*, 25(3): 51–62.

13. M. Williams. June 11, 1992. Interview of J.N.P. Hume. Transcript provided by Michael Williams.

14. S. M. Campbell. 2009. Backwater calculations for the St. Lawrence Seaway and the first computer in Canada. *Canadian Journal of Civil Engineering*, 36(7): 1164–1169.

15. News: Association for Computing Machinery. 1953. Program, ACM52 in Toronto, Mathematical Tables and Other Aids to Computation 7(41): 57–60.

Canadian contributions to the ACM meeting were modest. One staff member, who had developed the input-output element of Toronto's electronic computer, gave a talk about the machine explaining its design, capabilities, and their limited experience using it.[16] The ACM attendees were also given a tour of the Computation Centre facilities. Aside from the Ferranti Mark I and Toronto's own electronic prototype, this included a well-used IBM 602A electromechanical calculator, a rarely-used Meccano differential analyzer based on Douglas Hartree's 1933 design, and a handful of desktop calculators.

Did the ACM have any lasting connection or influence in Canada after the 1952 meeting? In the decades following, the ACM would shed its early emphasis on computing "machinery" to become the largest academic computer science society in North America. Toronto was among the first universities to drop its hardware development program to focus on computing applications and the roots of what would become computer science. This early transition might have put it at the leading edge of the ACM's efforts, but this did not happen. Only six publications by Canadian authors can be found in either *Journal of the ACM* (*JACM*) or *Communications of the ACM* (*CACM*) in the 1950s. Greater Canadian contribution and leadership within the organization only came in the following decade, in particular when Kelly Gotlieb was named the Editor-in-Chief of *Communications of the ACM* from 1962–1965, and of *JACM* from 1966–1968.[17]

Given the relative lack of large- or small-scale computing machinery in Canada dedicated to scientific or mathematical purposes, there were very few Canadians who could write an article suitable for the ACM. There were also established research journals such as *Physics in Canada* and the similarly named but wholly different *Canadian Journal of Physics*, more appropriate for junior faculty members looking to publish and advance their careers. It's also possible Toronto was too isolated with the only Ferranti Mark I in North America, at least until it purchased a more common IBM 650 in 1958. Nonetheless, the ACM did not return to Canada for a conference for many years and lost its early-entry advantage: by the late 1950s, several other computing-related organizations appeared, both within Canada and from the U.S., offering their own regular meetings and periodicals.

16. R. F. Johnston. 1952. The University of Toronto Electronic Computer. *ACM '52: Proceedings of the 1952 ACM National Meeting (Toronto)*, pp. 154—160. ACM Press.

17. Gotlieb became an ACM Fellow in 1994, and for 20 years he served as chair or co-chair of the ACM Awards Committee. He is also Fellow of the Royal Society of Canada and the British Computer Society, among other honors.

Figure 8.1 **Kelly Gotlieb (long-time ACM Awards Committee chair) presents 1990 Turing Prize to Fernando Corbató for his "pioneering work [on] CTSS and Multics."** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

8.3 Canadian Computing and Data Processing Societies

The Computing and Data Processing Society of Canada (CDPSC) was formed in the immediate aftermath of a successful national computing conference at the University of Toronto in June 1958. The conference had been in the works for at least two years, after a proposal for a Canadian computing organization surfaced in late 1955.[18] With only three general-purpose electronic computers in the country—all dedicated to scientific computation—the Computation Centre staff at Toronto, who were seen as the nation's computing leaders, were initially reluctant to support the effort.[19]

18. E. F. Codd to C. C. Gotlieb. December 29, 1955. University of Toronto Archives, B2002–0003, Box 2, Folder 3.

19. C. C. Gotlieb to E. F. Codd. January 4, 1956. University of Toronto Archives, B2002–0003, Box 2, Folder 4.

However, by the fall of 1957, they agreed that there was a "sharp increase in the number of Canadian companies committing themselves to electronic computers and data processors, and the feeling has grown that a conference of Canadian users of such equipment would have much to offer."[20] The number of computers was doubling year after year, and diversity of use was increasing. Canadians were also joining computing societies and attending conferences in the U.S. and Europe, but perhaps there also were sufficient problems of particular interest to Canadians to warrant a national organization.

Nearly 400 people registered for the 1958 computing conference. Their motivations appear to have been split roughly between scientific computation and data processing, and the talks ranged from general introductory surveys of the technology to highly scientific and technical work of advanced specialists.[21]

In the aftermath of this successful event, several attendees followed through by establishing a national computing society. Like the conference, the society offered a big tent to welcome almost any Canadian with an interest in computing and sought to promote "the advancement of computing and data processing."[22] Subsequent biennial conferences in 1960 and 1962 reflected these broad aims. The proceedings always included a mix of talks on scientific computing, data processing, and matters of general interest.[23]

In 1961 the society also began publishing the *Quarterly Bulletin*, "intended as a medium for the exchange of information concerning Society affairs and news items about the field of computing and data processing which are of interest to Canadian readers."[24] Many early issues were short, some less than 16 pages—simple text on inexpensive paper. It was the only such Canadian publication and, aside from general news, included updates from the society president, various committees, and local chapters across Canada. The occasional scientific article appeared in its

20. C. C. Gotlieb letter. September 12, 1957. University of Toronto Archives, B2002–0003, Box 2, Folder 2.

21. *Canadian Conference for Computing and Data Processing*, Proceedings. 1958. University of Toronto Press, Toronto.

22. R. L. Sutton, editor. October 1964. *Computing and Data Processing Society of Canada Quarterly Bulletin*, 5(1).

23. *2nd Canadian Conference for Computing and Data Processing*. June 6–7, 1960. Computing and Data Processing Society of Canada, University of Toronto Press, Toronto; and *Proceedings of the Third Conference of the Computing and Data Processing Society of Canada*. 1963. McGill University. June 2–3, 1962. University of Toronto Press, Toronto.

24. J. H. Aitchison. 1962. *Computing and Data Processing Society of Canada Quarterly Bulletin*, 2(4).

pages, but the *Quarterly Bulletin* never quite achieved the prestige or critical mass of an academic journal.

The CDPSC was also the Canadian representative to the International Federation for Information Processing and a charter member. IFIP was created to establish a worldwide forum for computer specialists, typically those with roots in math and engineering, and to advance the state of the art in computing-related fields. Five Canadians, including Kelly Gotlieb of the University of Toronto, attended the preliminary International Conference on Information Processing in Paris in June 1959. The Federation was then established in 1960 and the first official IFIP Congress took place in Munich in August 1962. Seven papers from Canada were accepted, on topics ranging from numerical computation to the use of computers in traffic control and medical research.[25] CDPSC members also sat on IFIP's early Technical Committees: TC-1 Terminology and Symbols, TC-2 Programming Languages, and TC-3 Education.

The connection to IFIP helped to drive the CDPSC toward more serious considerations of computing education in the mid-1960s. For example, Canadian TC-3 representatives were required to investigate and report on standards, qualifications, educational opportunities, and links to academic institutions. This in turn encouraged greater self-reflection. Many questions were raised about the future of the Canadian computing professional and the role of the CDPSC: What was the appropriate structure or content of computing courses for members? Should the CDPSC emphasize academic fields such as computer science and numerical analysis or practical studies in the application of computers to business? Who was qualified to teach such courses, and should they be taught exclusively at universities or also at private schools? And the ultimate question: what qualifications were necessary for practitioners, particularly for those without university degrees?[26] As in the U.S., these questions remained vexing ones well into the 1970s.

By the mid-1960s, there were over 1,000 CDPSC members in chapters across the country. To help address the problem of continuing education, a special committee began organizing tours across Canada by experienced lecturers, who offered

25. IFIP Munich 1962 Program. Charles Babbage Institute, International Computing Collection CBI 62, Box 12, Folder 1; and A. O. Downing, editor. 1964. *Computing and Data Processing Society of Canada Quarterly Bulletin*, 4(4): 10–14.

26. Anonymous. 1965. Progress reports from committees. *Computing and Data Processing Society of Canada Quarterly Bulletin*, 5(2): 7–8.

educational talks on a variety of topics.[27] In 1966, a new French-speaking chapter in Montreal, known as "L'Association Canadienne de l'Informatique," was created and the society became (nominally) bilingual.[28] In 1967, the *Quarterly Bulletin* was revamped, becoming larger and gaining a glossy format with letterpress printing. By then, it was being mailed to destinations that included the United Kingdom and Australia.[29]

Members also started to encourage the society to become more proactive by going beyond internal educational development to external advocacy. A letter to the editor in October 1965 encouraged members, local chapters, and the national organization to lobby the federal government to switch to the metric system of weights and measures. When this motion was put forward the following year, it received almost unanimous support.[30] Other members began to express concerns about information privacy, trans-border data flow, and Canadian import tariffs on computing hardware.[31]

Two articles in the *Quarterly Bulletin* in 1965 and 1969 brought many of these outward-thinking ideas together by exploring social responsibilities and computing professionalization. Leslie Mezei—a computer scientist at the University of Toronto with humanist and artistic interests in the use of computers—encouraged an ethical approach to the development of computing applications, via a code of ethics if necessary. He also reminded members that a desire to be treated as professionals and to update an undesirable public image as mere servants of machines had to be matched by their upholding their own responsibilities to consider the problems of automation and employment, privacy and human values, and to use technology for social, not only economic, improvement.[32]

Before the decade was out, the society had changed its name three times. In June 1965, it was temporarily renamed the Computer Society of Canada. The shorter

27. Anonymous. 1964. Visiting lecturers. *Computing and Data Processing Society of Canada Quarterly Bulletin*, 5(1): 9–13.

28. J. W. Graham. 1966. Message from the President. *Computer Society of Canada Quarterly Bulletin*, 6(4): 3.

29. Anonymous. 1966. *Computer Society of Canada Quarterly Bulletin*, 7(1): 2.

30. J. B. Reid. 1965. CSC and the metric system. *Computer Society of Canada Quarterly Bulletin*, 6(1): 21.

31. Observer. 1967. Bits and bytes. Spring 1967. *Computer Society of Canada Quarterly Bulletin*, 7(3): 26; Ibid., 7(5): 37.

32. L. Mezei. 1965. The social responsibility of computer specialists. April 1965. *Computing and Data Processing Society of Canada Quarterly Bulletin*, 5(3): 32–43, 51; and L. Mezei. 1969. Social responsibility. *Information Processing Society of Canada Quarterly Bulletin*, 9(2): 5–6.

name was part of an effort to incorporate and limit the liability of executives, who were simply taking the opportunity to choose an even more inclusive name, as suggested by the Policy Committee.[33] In 1968, the ongoing incorporation efforts led to a new choice: the Canadian Information Processing Society (CIPS), which, conveniently, offered better alignment with IFIP.[34]. This name was rejected by the Canadian government, which contended that the prefix "Canadian" was restricted to federal institutions. A third name—the Information Processing Society of Canada—was then used temporarily until the government reversed its decision the following year and CIPS—the second choice—became the name it has kept to this day.[35]

Whatever its name, CIPS continued to grow. By 1970, it had over 1,700 members and was the self-proclaimed "industry spokesman" in Canada.[36] Speaking with one voice is a key characteristic of professionalization, but several other noteworthy organizations, with different perspectives and interests, were competing for the same standing and prestige.

The Computer Science Association (CSA) was created in 1964 to accelerate "the development of computer science in Canada, by having as a primary objective the improvement of academic programs in universities."[37] Computer science as a discipline was already recognized at several major universities across Canada: in 1962 the University of Toronto renamed the Computation Centre to The Institute for Computer Science; in 1963 both the University of Alberta and the University of Western Ontario began advertising undergraduate computer science degrees starting the following year; in 1964 the University of Waterloo added its own computer science specialization within the mathematics department, and the University of Toronto created the first graduate department of computer science in Canada. The CSA was open to any individual or institution involved in teaching computer science or conducting related research, making it smaller and more specialized than CIPS. It held two successful conferences in the mid-1960s, and created a small journal. However, its relatively small size and a considerable overlap in membership

33. Anonymous. 1965. *Computer Society of Canada Quarterly Bulletin*, 5(4); and J. W. Graham. 1966. Message from the President. *Computer Society of Canada Quarterly Bulletin*, 6(3): 5.

34. R. P. B. Editorial. Summer 1968. *Canadian Information Processing Society Quarterly Bulletin*, 8(5): 2

35. B. B. "Ike" Goodfellow. 1968. President's message. *Canadian Information Processing Society Quarterly Bulletin*, 8(6): 2.

36. M. Kutt. 1970. President's message. *CIPS Mag.*, 1(1): 9.

37. *Computer Science Association Newsletter*. March 1965. 1(3).

between CIPS and CSA led to a merger in 1969, with the CSA becoming a special-interest group in the much larger organization.[38]

The Canadian Operations Research Society (CORS) was founded in 1957 and so was slightly older than CIPS. It drew together the Operational Research Group of the Defence Research Board, "with its affiliated branches in the various armed services" and the Operations Research Society of Toronto, plus a handful of key Canadian figures in the field.[39] Wanting to publish practical, academic-quality articles, it launched *CORS Journal* in 1963.

CORS was never a computing society, but interests with CIPS members overlapped enough that, in the late 1960s, the two societies began negotiating ways cooperate. In particular, CIPS had been hoping to introduce a formal and prestigious publication to supplement the more magazine-like *Quarterly Bulletin*, but rather than go it alone, in 1971 it partnered with CORS to create *INFOR: The Canadian Journal of Operational Research and Information Processing*. The new journal replaced *CORS Journal* and continued to provide technical articles aimed at a scholarly audience. However, its scope was expanded to include operations research and "the practice and theory of information processing" to best serve a perceived "inevitable drawing together of these two branches of applied science in the future."[40] The *Quarterly Bulletin* was renamed *CIPS Magazine* around the same time, and continued to publish lighter news and opinion articles.

8.4 The DPMA in Canada

For the entire life of CIPS, another, larger computing society was active in Canada: the Data Processing Management Association, or DPMA. It was rarely mentioned in the pages of the *Quarterly Bulletin,* although in the late 1960s and early 1970s the two societies competed as promoters of computing professionalization in Canada. One critical difference is that the DPMA (from 1951 to 1962 known as the National Machine Accountants Association or NMAA), like the ACM, was an American-based organization that permitted Canadian members.[41]

38. Anonymous. 1970. Bits of information. *CIPS Mag.*, 1(8): 6.

39. P. J. Sandiford. 1963. The origin and growth of the Canadian Operational Research Society. *Journal of the Canadian Operational Research Society*, 1(1): 1–13.

40. K. Radford. 1971. Editorial. *INFOR Journal*, 9(1).

41. For a deeper history of the DPMA, see T. Haigh. 2002. Technology, information and power: Managerial technicians in corporate America, 1917–2000. Ph.D. dissertation, University of Pennsylvania, Philadelphia.

Figure 8.2 **National Machine Accountants Association (NMAA) connected DPMA in Canada as elsewhere with the early "punched card" computing tradition.** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

The DPMA grew quickly in the 1950s, with a widely decentralized structure that gave a great deal of autonomy to its many local chapters across the U.S. and Canada. The first Canadian chapter in Montreal was admitted in 1953, followed by Toronto in 1955, and Calgary, Winnipeg, and Vancouver in 1956. Not only were Toronto and Montreal the two largest cities in Canada, they were also the home to many corporate headquarters and government offices, and logically the home to the most potential members. In fact, the Toronto chapter was regularly among the largest in the entire association. In 1958, the DPMA had over 11,000 members and the Toronto chapter had 234; only Boston, Chicago, Los Angeles, and New York had larger chapters. From the mid-1950s through the 1960s, around 5% of the overall membership was Canadian.[42] The Canadian membership numbers, while relatively small within the DPMA, were still larger than the total CIPS membership at any given time.

Several Canadians joined the DPMA Executive Committee or served in high-ranking positions. In 1958, Donald T. Barber, a management consultant in Toronto, was elected Vice-President, Library and Publications, a position he held until 1961. In 1964, Daniel A. Will, a senior analyst in the data processing operations at Canadian National Railway in Montreal, was elected President, after serving since 1961 as Vice-President of Education.[43]

It is not clear that Canadians or their interests received special consideration in the DPMA. In the 1950s, Canadian chapters that joined the NMAA were typically inserted into nearby American regional divisions. A February 1960 proposal to reorganize the divisions merely repeated and extended this practice, despite a 1959 resolution from the Executive Committee that it "ascertain the wishes of chapters including those outside of the United States on the current thinking of the advantages that accrue from a change of the Association name and emblem."[44] However, a year after holding its national meeting in Toronto in 1961, it adopted a superficially more inclusive name, becoming the Data Processing Management Association. It also renamed the annual meeting the "International Conference."[45]

What did Canadians think of their relative obscurity within the DPMA? Several inconspicuous events shed some light.

When in late 1967 the DPMA headquarters in Chicago shifted to a centralized billing system, it ran into resistance from Canadian chapters. Members were used to paying combined local and national dues to the local chapter, which forwarded the national portion to Chicago. The new system required that individual members pay national dues directly to headquarters. Almost immediately, complaints

42. NMAA Board of Directors Meeting. February 1953. Data Processing Management Association Records, Charles Babbage Institute, CBI 88, Box 1, Folder 2; NMAA Board of Directors Meeting. June 1955. CBI 88, Box 1, Folder 7; NMAA Board of Directors Meeting, June 1956, CBI 88, Box 1, Folder 12.

43. Daniel Will biography, c. 1964. CBI 88, Box 31, Folder 48.

44. NMAA Board of Directors Meeting. February 1960. CBI 88, Box 1, Folder 30; NMAA Board of Directors Meeting. February 1959. CBI 88, Box 25, Folder 1.

45. DPMA Report. February 1962. CBI 88, Box 1, Folder 39.

were made to the Executive Committee about the inconvenience and costs of international money orders and paying in U.S. funds, and the failure to account for Canadian circumstances; a decline in membership renewals followed. As George Fawkes, International Director of the Vancouver Chapter, said: "We have felt very strongly about this matter of paying our dues in US funds, but since it is in the by-laws and you Damn Yankees want your pound of flesh, we will abide by it."[46] Although the DPMA Treasurer initially claimed that the new billing system simply couldn't accommodate Canadians, noting that "a system can only stand so many exceptions," eventually he created such an exception and opened a Canadian bank account to facilitate the paying of dues.[47]

As well, several times during the latter half of the decade, Canadians complained to the DPMA executive about what seems the most trivial of matters: stamps. Mailings from headquarters to the membership or to various chapter officials often included self-addressed postage-paid envelopes for replies or payment of dues. Of course, U.S. stamps or marks were of no use in Canada and Canadians were forced to buy stamps to participate in DPMA affairs. Don Jervis of Hamilton, Ontario and president of the regional DPMA Seaway Division (primarily New York, Ontario, Quebec, and the Atlantic provinces) pointed out that even his cousin in Detroit had thought to include appropriate stamps when sending out wedding invitations to relatives across the border, adding, "it is not the few pennies for postage that concerns me, but rather the good will that could be generated by a very small amount of effort in this situation."[48] R. C. Elliott, the long-time Executive Director of DPMA, could only admit that the mistake had happened because "no thought was given to the Canadian member," although this was not the first or last time he received such a complaint.[49]

Collectively, such issues became known to the DPMA executive as the "Canadian Problem." They seemed to surface about once a year and were typically easy to fix, but remained an ongoing and apparently puzzling issue to the Americans. Poor handling of the Canadian Problem eventually contributed to the creation of the semi-independent Canadian Data Processing and Management Association Institute (CDPMAI) in 1971 by a fractious group of Canadians. Much of the

46. From context, it's clear the epithet was tongue-in-cheek, but the overall sentiment was serious. G. Fawkes to D. B. Johnston. October 2, 1967. CBI 88, Box 18, Folder 17.

47. Memo from R. E. Kuempel. October 24, 1967. Canadian checks & bank accounts. CBI 88, Box 18, Folder 17.

48. D. J. Jervis to R. C. Elliott. October 13, 1971. CBI 88, Box 19, Folder 42.

49. R. C. Elliott to D. F. Jervis. July 22, 1971. CBI 88, Box 19, Folder 42.

dissatisfaction of this particular group can be traced to the "Toronto magazine controversy" of 1969.

For many years, DPMA chapters and regional divisions were encouraged to produce newsletters containing announcements, articles, and notices of events of local interest. In 1969 members of the Toronto chapter, working in concert with four nearby chapters, started work on *DPMA Magazine*, an exclusively Canadian publication with Canadian advertisements (normally excluded from the DPMA's official *Journal of Data Management*), news, articles, and contributions from each of the 14 Canadian chapters. Although the Toronto organizers believed that they had obtained permission from the DPMA executive a year earlier, when detailed plans for the magazine came to light in the spring of 1969, the matter erupted into a full-blown conflict between the Toronto chapter and the Chicago executive.

The Toronto chapter and particularly its International Director, Len Turner, initially saw the magazine as a wholly approved regional inter-chapter newsletter, which did not seem to break any rules. However, at some point, they expanded the debut issue into a Canada-wide magazine, which was to be added to the registration kit given to all attendees (Canadian and otherwise) of an upcoming DPMA conference in Montreal. After a review that lasted months, the executive declared that since the Canadian magazine would be competing unfairly with the DPMA's own periodical, the *Journal*, the Canadian production had to stop immediately.

After heated discussions and careful negotiations that employed semi-neutral intermediaries, the two sides reached a compromise: one issue of the magazine could be printed for the Montreal conference, but further issues were forbidden.[50] In compensation, Elliott offered Turner a monthly "Canada" column in the soon-to-be revamped *Journal* to cover issues of interest to Canadians.[51] As the new editor noted, the typical issue featured articles about legislation, standards, news, and new products that could all benefit from the addition of a Canadian perspective.[52] There had been virtually no Canadian content in the previous year's issues; this situation may help to explain the Toronto chapter's original magazine plan.[53]

50. L. Turner. September 1969. Report on publication policy to the Toronto chapter DPMA. CBI 88, Box 18, Folder 20.

51. R. C. Elliot. Memorandum. September 29, 1969. CBI 88, Box 18, Folder 20.

52. D. C. Kitzmiller to L. Turner. September 26, 1969. CBI 88, Box 18, Folder 20.

53. One exception to the Canada-lite content came in the July 1969 issue: an advertisement for a luxury set of leather bound, gold-embossed DPMA *Journal* back-issues, complete with space-age bookends. The accompanying photo featured a second shelf with just one slim, unexplained, and similarly bound book, entitled "Canada: Sea to Sea" and positioned just slightly "north" of the *Journal*.

In spite of the new column, the magazine controversy remained an open wound for Turner and several other high-level Canadian DPMA members. They dredged up the fiasco repeatedly for several years, to the exasperation of the Executive Director, for whom the Canadian Problem simply would not go away.

By 1971 tensions reached a tipping point, with the Toronto chapter prepared to leave the DPMA. It did not help that CIPS had just relaunched the *Quarterly Bulletin* as a glossy monthly magazine that appeared to copy the format of the cancelled Canadian DPMA magazine, and that when *Canadian Datasystems* (similar to *Datamation*) had recently been launched, the DPMA Executive Director had publicly congratulated the new publisher. But the main schism-creating wedge was not a magazine but accreditation.

8.5 CIPS, DPMA, and the Canadian Accreditation Battle

As historian Nathan Ensmenger has explained, accreditation was critical to computer professionalization in the 1960s. Various groups contended over questions of who should be accredited, how, and why.[54] Was accreditation necessary for academics and computer scientists, for corporate systems analysts and programmers? There was a general agreement that accreditation should test and verify a certain body of knowledge, but which fields were "core" and how many years of study were necessary? And who benefited from accreditation? Employers, looking for a stamp of quality? Potential employees, looking for an edge in getting hired or promoted? Or was it largely the accrediting organization—able to justify its existence as the ultimate gatekeeper to an elite profession? These questions remained topics of vigorous debate well into the 1970s and 1980s, and important parallels can be seen in the development of computer science in this period.[55]

The DPMA provided the earliest accreditation to computing professionals—in Canada and the U.S. Its Certificate in Data Processing (CDP) was created in 1962 as "the first attempt by a professional association to establish rigorous standards of professional accomplishment in the data processing field."[56] Although the CDP was accepted by the computing field initially, the DPMA struggled throughout the decade to maintain momentum and avoid controversy. In 1970, an attempt to require that candidates for the CDP possess a bachelor's degree ignited a crisis in

54. N. L. Ensmenger. 2012. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*. MIT Press, Cambridge, MA. See Chapter 7, The professionalization of programming, pp. 163–194.

55. See Dziallas and Fincher's chapter 5 in this volume.

56. N. L. Ensmenger. 2001. The 'question of professionalism' in the computer fields. *IEEE Ann. Hist. Comput.* 23(4): 11.

the U.S. While a degree was generally an industry hiring requirement, many existing CDP holders lacked post-secondary qualifications.

Around the same time, accreditation became a key concern of Canadian efforts in professionalization. In the late 1960s, a CIPS accreditation committee started to study the problem as part of an overall plan to work more closely with government on many aspects of the computing industry.[57] As the committee indicated in its first report, published in the spring of 1969, the broad objectives of a professional body in Canada would be to establish standards of practice and a code of ethics to protect both the public and the industry, and to ensure "effective education and systematic apprenticeship" opportunities for practitioners.[58] To achieve these objectives, the committee proposed an independent institute headed by a council, with representatives from relevant groups, including CIPS, DPMA, CORS, and CSA. The council would establish committees and subsequent standards for accreditation, education, ethics, and public relations. The report also suggested several topics and fields of study to form core and specialist bodies of knowledge, as well as standards of professional conduct.

This umbrella arrangement of a council representing the various societies and organizations was similar to IFIP and the American Federation of Information Processing Societies (AFIPS), but had also been proposed elsewhere and earlier in Canada. In 1968, the federal government's Central Data Processing Service Bureau had suggested to the Ottawa Valley DPMA chapter that the DPMA, CIPS, and CORS form a Canadian Federation of Information Societies (CFIS) to better represent their joint interests and share resources.[59] The president of the Ottawa Valley DPMA chapter admitted he was only able to comment on behalf of his chapter, not all Canadian DPMA members. This was a key difference between CIPS and the DPMA: the former could speak with one voice, through the elected national executive; the latter was divided into independent local chapters and regions, without a single, national voice in Canada.

Unencumbered by an American parent organization, CIPS pushed ahead on collaboration and accreditation, working closely with CORS to host a joint conference in Vancouver and produce the new journal *INFOR*, hosted a national computer exhibition in Montreal, and continued to sell its accreditation proposal to critics.[60]

57. B. B. "Ike" Goodfellow. 1969. President's message. *Information Processing Society of Canada Quarterly Bulletin*, 9(2): 3–4.

58. Supplement. 1969. *Information Processing Society of Canada Quarterly Bulletin*, 9(2).

59. K. J. Radford to A. Laroque. July 16, 1968. CBI 88, Box 18 Folder 19.

60. M. Kutt. President's message. 1970. *CIPS Mag.*, 1(1): 20.

Unfortunately, even the most fundamental issue—defining an information processing profession—remained problematic. Was a professional designation intended for the relatively small number of "elite" computer scientists, for the larger number of day-to-day data processing managers and staff, or should it expansively include almost anyone involved in computing?[61] As the accreditation committee chair noted, finding consensus was time-consuming and frustrating: "I can now sympathize with the people that tried to build the Tower of Babel."[62] Relations between CIPS and DPMA, always a little competitive and antagonistic, never reached the necessary level of cooperation. References to "foreign dominated" computer societies appeared in the *Quarterly Bulletin;* letters between DPMA members often made similar veiled and unfriendly references to the "other" organization.[63] A 1971 CIPS policy to cooperate "actively with national or international organizations in the information processing field" made references to both the ACM and the British Computing Society, but failed to note the DPMA in Canada.[64]

In 1970, George Fierheller, newly elected president of CIPS, reiterated the view that CIPS was the only legitimate Canadian computing society, as it was not a small part of a larger American organization.[65] At the time, just a few years past the celebrations of the national centenary in 1967, Canadians were increasingly sensitive to criticism that they were slow to adopt technology, unable to exploit their own creativity, and possessed of an economy that was nothing more than a "branch-plant," effectively subservient to the whims of American manufacturers.[66] Fierheller and others held that Canadians should be able to manage their own affairs—including computing—and should not accept being treated "as the 51st state."[67]

The Canadian ranks of the DPMA saw the CIPS proposal as a clear and present threat to their own organization and the CDP, made stronger by the improved reputation that CIPS now appeared to enjoy across the country. But how to

61. B. P. O'Connel. 1969. Letter to the editor. *Quarterly Bulletin*, 9(4): 42; and Len Turner. 1970. Professionalism: Curse or course. *CIPS Mag.*, 1(4): 21.

62. W. Kerrigan. 1971. Excerpt from the speeches. *CIPS Mag.*, 2(2): 16.

63. Anonymous. 1967. Bits and bytes. *Computer Society of Canada Quarterly Bulletin*, 7(5): 37.

64. G. A. Fierheller. 1971. President's message. *CIPS Mag.*, 2(2): 4.

65. G. A. Fierheller. 1970. President's message. *CIPS Mag.*, 1(8): 5.

66. One of the most notable books to take up this argument is J. J. Brown. 1967. *Ideas in Exile: A History of Canadian Invention*. McClelland and Stewart Ltd., Toronto.

67. B. B. "Ike" Goodfellow. 1970. Past President Ike Goodfellow agreed with the proposal in principle. *CIPS Mag.*, 1(11): 7.

respond? Doing nothing while hoping that CIPS would fail was risky; petty bickering was unprofessional. Cooperating with CIPS to build a new Canadian accreditation standard could be the best option, if the American executive of the DPMA would authorize a single Canadian DPMA voice.

A senior Canadian DPMA member sent a letter to the executive, past presidents (including Canadian Daniel Will), and other high-level Canadians in the organization, inviting comments on a proposal to create an official executive position that could provide "A Canadian Voice."[68] As politely as possible, he explained that the motivation was not "the Canadian Problem" or even better representation of Canadian interests within the DPMA. Instead, he intended this new executive member to have authority to interact with Canadian government agencies, corporations, educational institutions, newspapers, and other organizations that were now turning to CIPS for advice and recommendations—and, of course, to work with CIPS on the accreditation problem.

The Americans on the DPMA executive were generally sympathetic in acknowledging that Canadians might have unique concerns, but many doubted the necessity of a special executive position. DPMA President J. D. Parker agreed that having Canadians work together on problems of accreditation and government interaction was best, but admitted that his lack of attention to the Canadian Problem was now handicapping his ability to respond effectively.[69]

DPMA Education Director Don MacPherson travelled to Toronto several times as a special envoy. Portions of his reports in which he evaluates the CIPS plan and the Toronto DPMA chapter's ability to negotiate on behalf of the larger organization take on near-paranoid tones. Was CIPS acting in good faith or was it scheming to take over the CDP and do away with the DPMA in Canada? Could the Toronto DPMA chapter be trusted to do the right thing? But, correctly and critically, he urged the DPMA executive to provide direction immediately; otherwise this particular Canadian problem would soon grow beyond their control.[70]

Ultimately, the lackluster response by the American DPMA executive to accept a "Canadian voice," lingering distaste over the magazine fiasco, and desperate fear that CIPS would move ahead unilaterally, all combined to unite the leading Canadian DPMA members. At a special meeting in Quebec City in September 1971, they voted to create the Canadian DPMA Institute (CDPMAI)—the desired single Cana-

68. D. F. Jervis to J. D. Parker. April 22, 1971. CBI 88, Box 18, Folder 19.

69. J. D. Parker to D. F. Jervis. March 26, 1970. CBI 88, Box 22, Folder 10.

70. D. MacPherson to R. C. Elliott. March 11, 1971 and October 1, 1971. CBI 88, Box 22, Folder 21.

dian voice. Clearly designed to compete with CIPS and its accreditation proposal, the Institute made immediate plans to adopt the CDP as the definitive accreditation program in Canada. As it launched efforts to build links to various federal and provincial government agencies, CDPMAI continued to deny any ulterior motives or intentions to split from the parent DPMA organization. Its aim was simply to deal with "matters of interest and concern to the members of the Canadian Information Processing Community."[71]

Len Turner, the CDPMAI's founding chairman, explained the creation of the CDPMAI in his next "Canada" column in the DPMA *Journal*. He enthusiastically described the efforts as the splitting off of "Caesar's Gaul" to create proper Canadian representation within the DPMA, rather than "Canadian chauvinism" at work.[72] Unsurprisingly, the executive were less enthusiastic. Executive Director Elliott noted that Turner "almost sounds like Karl Marx in his beginning days."[73] DPMA President E. O. Lineback more tactfully suggested to Turner that a "more positive approach" would be more beneficial, and downplayed Turner's anguished recounting of the three-year-old Toronto magazine controversy.[74] But despite the lack of support south of the border, the CDPMAI felt ready to compete with CIPS on the accreditation battlefront.

Ironically, the CIPS plan for professionalism and a Canadian accreditation council seemed to be falling apart. Two position papers, arguing for and against accreditation, appeared in *CIPS Magazine* in December 1971. On the "yes" side was William Kerrigan, chair of the accreditation committee for several years. He argued for its necessity on the basis of taking responsibility and independent self-regulation (instead of government-imposed), while admitting that many of the detailed questions about accreditation processes, standards, training, costs, and benefits remained unanswered.[75] On the "no" side was industry leader Ian P. Sharp, who argued against any formal accreditation programs because, in his view, professionalization existed primarily to protect the public and "the main bulk of the work carried out by computer people is in the form of services to general management," not the public.[76] A slew of letters published in that same issue seemed to support

71. L. Turner to CDPMAI Council Members. October 10, 1971. CBI 88, Box 22, Folder 9.

72. L. Turner to Donald A. Young. November 1, 1971. CBI 88, Box 19, Folder 42.

73. R. C. Elliott. November 9, 1971. Memorandum. CBI 88, Box 19, Folder 22.

74. E. O. Lineback to L. Turner. January 4, 1972. CBI 88, Box 22, Folder 9.

75. W. Kerrigan. 1971. Accreditation?—Yes. *CIPS Mag.*, 2(9): 8.

76. I. P. Sharp. 1971. Accreditation?–No. *CIPS Mag.*, 2(9): 9, 14.

the "no" side; curiously, they generally ignored the problem of public reputation of their field.[77] Instead, many noted that regulation of any kind would only slow the industry down, "hindering" development.[78] Even those in favor of accreditation generally admitted that there were too many unanswered questions related to certifying a diverse computing profession.

At a February 1972 meeting of the CIPS national board, just a few months after the creation of the CDPMAI, CIPS declared publicly it would drop accreditation as an immediate priority. A Canada-wide poll of CIPS members in late 1971 showed 475 against accreditation, 349 in favor, and 52 undecided (although there were some accusations of ballot-stuffing).[79] Yet behind the scenes, efforts by the CIPS executive continued. In particular, without a great deal of fanfare, it joined with the ACM and DPMA in the U.S. to create the Computer Foundation, a non-profit organization to administer and manage computing accreditation. The foundation was renamed the Institute for the Certification of Computer Professionals (ICCP) when it was officially established in August 1973 as an umbrella-like accreditation council for the U.S. and Canada. The ICCP eventually adopted the DPMA's CDP as the foundation of its accreditation plan, and so, almost through the backdoor, CIPS would come to adopt the CDP as well.

In 1974, the CIPS executive, somewhat quietly and without a general membership vote, accepted the ICCP as the preferred accreditation standard for its members. The CDP was now endorsed for a major of Canadian and American computing professionals via CIPS and CDPMAI, as well as ACM and DPMA.[80] Unfortunately, industry confidence in the certification process remained low, and the problems of accreditation remained unsolved. Questions about professionalism and qualifications and the value of the ICCP in the computing field continued to be posed through that decade. CIPS and the DPMA were uneasy allies at best, only occasionally able to cooperate or collaborate effectively, and accreditation rarely reached the earlier heights of concern.[81]

77. As Dziallas and Fincher (chapter 5 of this volume) point out, public reputation was a driving concern for Curriculum '68, the model computer science curriculum developed by the ACM in 1968.

78. L. B. Moore. 1971. Unhindered. *CIPS Mag.*, 2(9): 12, 14.

79. Anonymous. 1972. National Board Report: CIPS members vote against accreditation. *CIPS Mag.*, 3(3): 9.

80. G. M. Pike. November 1973. ICCP–here's how it works. *CIPS Computer Magazine*, p. 11.

81. G. McInnes. 1977. President's report. *CIPS Review*, 2(1): 4.

8.6 Toward Canadian Identity?

The battle for Canadian accreditation was inspired by many of the same challenges as those found in the U.S. and led to similar results. Just as the ACM and DPMA and American industry struggled to define boundaries and structure for a computing profession, CIPS in Canada was involved with debates over academic qualifications and accreditation. Emphasis and control over education and training is, of course, a hallmark of professionalization, but educational efforts must be applied to a circumscribed area of work that distinguishes the profession. The Americans were unable to improve upon the CDP, and Canadian efforts bore no better fruit. As a result, the benefits of joining either country's computing societies remained unclear, and membership numbers failed to grow in proportion with the rapidly expanding industry.[82]

Technological conditions in Canada remained similar to those in the U.S. Most computing and data processing were carried out on the same mid-scale computers for the same reasons.[83] Although Canadians might have been a little slower to adopt and adapt the new technology, electronic data processing experienced rapid growth in Canada as in the U.S. And, as I have shown, Canadians were neither ignorant of nor out of touch with international developments in the field.

Given that Canadians, using the same technology for the same reasons, would end up with similar professional organizations, an interesting loose end to consider is the claim, made most often by CIPS, that a distinct Canadian approach was necessary.

Part of the Canadian identity is a need—occasionally a desperate one—to define that identity. Originally, in the 19th century this was typically framed as a question of what made a Canadian different from any other imperial British subject, but by the mid-20th century the framing became what made a Canadian different from an American. These debates have taken place through to today across the fields of literature, politics, scientific study, and technological enterprise.[84] Long, naive lists of "Canadian inventions" have been composed, but as historians of science and technology know, the effect of simultaneous discovery-and-invention can nullify such nationalist efforts. Several countries often claim the same ideas

82. Ensmenger, *The Computer Boys Take Over*, 189; G. A. Fierheller. 1970. President's message. *CIPS Mag*., 1(10): 6.

83. Jim Cortada pointed out to me that when he was an employee of IBM, the Canadian market was rarely, if ever, given special consideration.

84. See S. Zeller. 1987. *Inventing Canada*. University of Toronto Press, Toronto; and R. D. Francis. 2010. *The Technological Imperative in Canada*. University of British Columbia Press, Vancouver.

and inventions, but practitioners will borrow quite happily across borders. So why the resistance to borrow from American computer professionalization efforts?

An alternative view of technology and identity can be found in the work of Maurice Charland, who coined the phrase "technological nationalism" to explore how technologies can be granted a rhetorical power to build nations or develop a national consciousness. He observed, for example, nation-building language in the discourse surrounding the creation and growth of the Canadian Broadcasting Corporation (CBC), the national public radio and television broadcaster. In the 20th century, the CBC was often described as a unifying technology that carried a common Canadian culture to every corner of the country, thus reinforcing the state as well as a national awareness and identity.[85]

Robert MacDougall, in his exploration of the development of the Trans-Canada Telephone System, an all-Canadian, long-distance telephone network, extended Charland's work. He pointed out that technological nationalism can also be understood *ironically* in Canada. Normally, the rhetoric is a product of a "recurring hope that new communication technologies will not only nourish Canada's national identity, but also protect that identity from domination by the cultural and commercial engines of the U.S."[86] MacDougall shows how building complex national communication networks in response to such hopes can ultimately simplify the eventual *integration* of expanded Canadian and American technological systems and increase Canadian exposure to American influence.

In the case of computing professionalization in Canada, a similarly ironic pattern can be seen: nationalist claims requiring a distinct Canadian approach, followed by the acceptance of an American-designed standard.

Throughout the late 1960s, Canadian members of the DPMA struggled first to be recognized within their organization and then to establish a Canadian voice for the DPMA. This led to the creation of the CDPMAI. Its first chairman, Len Turner, argued in the pages of *Canadian Datasystems* that "Canadians don't want to be bottled up in a little society where they talk to each other and let the world go by. We

85. M. Charland. 1986. Technological nationalism. *Canadian Journal of Political and Social Theory*, 10(1): 196–220.

86. R. MacDougall. 2007. All red dream: Technological nationalism at the Trans-Canada Telephone System. In N. Hillmer and A. Chapnick, editors. *Canadas of the Mind: The Making and Unmaking of Canadian Nationalisms in the Twentieth Century*, p. 49. McGill-Queen's University Press, Montreal & Kingston. As MacDougall reminds us, historians have occasionally fallen prey to technological nationalism and embedded it within their own work. He cites Pierre Berton's depiction of the 19th century construction of a trans-national railway across Canada as the "National Dream" as one such case.

can contribute not only natural resources to the world but ideas and solutions to world needs. We want to be part of the International scene."[87] However, he also took pains to point out that the new institute did not represent a true split from DPMA. For example, the CDPMAI adopted a version of the DPMA logo with a Canadian leaf, and quickly moved forward with the American CDP as its chosen accreditation standard without customizing it to suit Canadian needs or perspectives.

During these same years, many articles appeared in CIPS magazines encouraging members of the Canadian computing industry to have more positive views of their own capabilities and identity, and to get past the Canadian stereotype of resistance to change and new technology. David Haltrecht asked for a strong Canadian element in the 1970 Canadian Computer Show and Seminar: "it took Expo '67 to bring out our national consciousness, to make us realize we have a Canadian identity, that we have the ability to do great things."[88] In 1972, Robert Perry, a Canadian newspaper editor, spoke derisively at a CIPS meeting about a Dutch technician who used English when working with an American-made computer, claiming that "no self-respecting American computer would speak Dutch."[89] In Perry's view, such submissiveness was wrong and technologies should be understood to have clear connections to national identity. Computing and data processing might well be the key to ridding the country of "a dependent mentality" in relation to the U.S. and to "creat[ing] a more exciting environment for its people." However, the proponents of Canadian professionalization and accreditation were rarely able to forge similar links to national identity. Meanwhile, some of the strongest resistance to accreditation came from Canadians who felt that the true obstacle to growth was not dominance from the U.S. but meddlesome regulations from their own national government. As John Vardalas has shown, during the 1960s and 1970s, the Canadian government generally demonstrated a poor understanding of the computer industry. Certainly, it failed to formulate a substantial policy to support the growth of domestic hardware manufacturers.[90] And as CIPS members in the early 1970s knew, the government was turning a regulatory eye to computer utilities and the information-processing industry as a whole with political purposes in mind.

87. L.Turner. November 1971. Canadian DPMA Institute formed to deal with domestic problems. *Canadian Datasystems*, pp. 36–37, 49.

88. D. Haltrect. 1970. Computer show/competition. *CIPS Mag.*, 1(7): 10. "Expo 67" was the successful 1967 World's Fair in Montreal, Quebec, and, because it was held the same year as the centenary of Canada, is often used as a shorthand for the national celebration that same year.

89. R. Perry. 1972. The American challenge: Stand on own feet, Perry tells Canada. *CIPS Mag.*, 3(6): 10–11.

90. J. N. Vardalas. 2001. *The Computer Revolution in Canada*. MIT Press, Cambridge, MA.

In a speech to CIPS members in February 1970, Minister of Communications Eric Kierans took up technologically nationalist language and warned of "a real danger that as a large computer utility market develops in Canada, it will be dominated by systems located south of the border. Should this trend be allowed to continue it could have grave implications for Canadian sovereignty."[91] His concern was over information control, trans-border data flow, and Canadian privacy—a fundamentally different situation than the resource-based, branch-plant economic problems of the past: "We are not simply losing one more manufacturing or service company, we are losing control of how information will be stored and processed."[92]

CIPS members responded with wary complaints that such discussion was more emotional than factual, and beliefs of their own (which were likely counter-factual) that telecommunication networks such as telegraphs and telephones had ignored the Canada-U.S. border for over a century. Regulations would be prohibitively difficult to implement and would slow Canadian innovation, putting growth at risk. Ultimately, CIPS itself came out against government regulation of computer utilities telecommunications and public data processing, arguing that the government lacked sufficient organization and study needed "to provide a precise definition of the real nature, the needs, and the rate of development of the whole information processing industry."[93] This was, ironically, the same criticism used by many CIPS members against accreditation: that it was an ill-defined and vague proposal with unclear goals that would slow growth.

Ultimately, the Canadian computer industry resisted government or self-imposed regulations of computer utilities or trans-border data flow, just as it did any serious attempt to require a recognized Canadian (or American) accreditation standard, no matter how well these ideas or regulatory possibilities may have served nationalist arguments of Canadian computing societies.[94]

---

91. C. Pickell. 1970. Editors message. *CIPS Mag.*, 1(2): 7.

92. D. R. McCamus. 1970. They worry too much. *CIPS Mag.*, 1(2): 21.

93. N. Erskine. 1970. A submission by the Canadian Information Processing Society related to the subject of participation by telecommunication carriers in public data processing. *CIPS Mag.*, 1(11): 10–13.

94. W. E. Cundiff and M. Reid, editors. 1979. *Issues in Canadian/US Transborder Computer Data Flows*. Institute for Research on Public Policy, Toronto.

9 The Anatomy of an Encounter: Transnational Mediation and Discipline Building in Cold War Computer Science

Ksenia Tatarchenko

Nowhere else in the Soviet Union did I find such a complete understanding of the American computer scene. It is really true that you are the Novosibirsk branch of the ACM.

—Edward Feigenbaum to Andrei Ershov (1964)

Spoken and written words do not always add up, even less so if put forth by a propaganda machine. "Soviet Expert on Soviet Units—Not Enough and Not Very Good," announced *Electronic News* on June 7, 1965, referring to a speech by Soviet computer pioneer Andrei Ershov to members of the Los Angeles chapter of the Association for Computing Machinery (ACM).[1] A lighthearted joke among specialists became a first-hand witness to Soviet failure. Pointing to a growing computer gap between the Soviet Union and the U.S., as well as to the subpar quality of Soviet hardware, the article's rhetoric was hardly remarkable in itself—comparison was a fundamental

1. R. Henkel. June 7, 1965. Too few and not good enough—Soviet specialist on Soviet Units. *Electronic News*, p. 33. Available at http://ershov.iis.nsk.su/archive/eaindex.asp?lang=1&did=26958.

feature of Cold War discourse defined by the competition for global dominance between the two superpowers. That the computer was among the subjects of choice for the American public discourse is no surprise: a strategic military technology, its circulation tightly controlled by a series of export embargoes, the ubiquitous media representations of the new machines were already structured by a techno-revolutionary narrative demonstrating American supremacy.[2]

While such considerations provide historical context, they are of less help to explain the notable encounter between Ershov and American experts. Multiple questions arise: What was a Soviet expert doing at the meeting of the largest chapter of one of America's key professional organizations in the field, located in a hub of the national military-industrial complex? And in a state that was home to what would soon become known as Silicon Valley? And, finally, who was this loose-lipped Soviet expert, and what was he trying to say? These questions must be answered in order to comprehend the nature of the event that took place at the ACM chapter meeting and the reactions to the *Electronic News* article by the Soviet specialist and his American hosts. I suggest that the article, misusing the Soviet expert's words to stress the inadequacy of the Soviet computer industry, was perceived as a double threat: to Ershov's career, but even more so to a shared international vision of computer science.

The main aim of Ershov's trip to the U.S. was to participate in the congress of the International Federation for Information Processing (IFIP) held in New York, May 24–29, 1965. In fact, the California leg of the trip was an add-on product of efforts on the part of the American community. When inviting Ershov to address the Los Angeles chapter, Paul Armer, its program chairman, specified that he did not expect a firm confirmation, as "a probability of one half of having you lecture to us is better than no chance at all."[3] A member of a 1959 delegation of American computer experts to the Soviet Union, Armer had already met Ershov and knew of his work on automatic programming. Moreover, he was well aware that since it was subject to approval by the Soviets and the U.S. State Department, any such trip was uncertain. Finally, Ershov had already been invited to Berkeley and Stanford, and

2. R. R. Kline. 2015. *The Cybernetics Moment: Or Why We Call Our Age the Information Age*. Johns Hopkins University Press, Baltimore.

3. Ershov Archive (hereafter **EA**): Paul Armer to Andrei Ershov. April 13, 1965. file (f.) 133/ list (l.) 112. The electronic version of the Ershov Archive for the History of Computing is the main source for this chapter. Available at http://ershov.iis.nsk.su/english/ (accessed March 2016).

the California specialists put together an impressive schedule including invitations to Berkeley, Stanford, UCLA, Caltech, and Armer's home institution, RAND.

When the visit actually occurred, the satisfaction appeared to be mutual. Armer wrote Ershov with hearty congratulations. His letter conveyed three points of view—that of the ACM chapter, RAND staff, and Armer's family who hosted Ershov. It opened with words of gratitude: "On behalf of the Los Angeles chapter of ACM, I thank you for your outstanding presentation to our meeting recently. It was a tremendous success. Our membership found your talk stimulating and most interesting."[4] Ershov responded with the appropriate expressions of recognition for all the organizational efforts.[5] But while this exchange of thank-you letters between the hosts and the guest was taking place, then came the tempest—the American media coverage of Ershov's presentation reached the Soviet Academy of Sciences, the institution responsible for overseeing scientific communications with abroad.

Conflicts and their resolutions are a historian's chance to glimpse insights into relationships that otherwise remain out of sight. In this chapter, I study the paper trail generated by Andrei Ershov's 1965 trip to the U.S., including his lecture at the ACM chapter. An exceptionally rich historical record, this trail outlines the interlocking geopolitical, institutional, and intellectual environments of the international computing community. I use the tools of micro-history to explore the practices of transnational mediation as integral to the process of discipline-building in the field of computer science.[6] The details pertinent to the organization of the trip, its media representations, and the traces of interactions with authorities reveal a set of interconnected relationships between experts, state, and society that illustrates the complex character of Cold War scientific communication. Drawing upon the studies of intercultural encounters, where accounts about experiences in other lands are often political manifestoes and forms of self-creation, I distinguish the multiple functions of a scientific encounter in the construction of professional identity.[7] I argue that the state "surveillance regime" was a two-way street allowing for the exercise of bottom-up agency: the security task of information collection

4. EA: Paul Armer to Andrei Ershov. June 21, 1965. f. 133/ l. 84.

5. EA: Andrei Ershov to Paul Armer. June 28, 1965. f. 126/ l. 155.

6. For "micro-history," see C. Ginzburg. 1980. *The Cheese and the Worms: The Cosmos of a Sixteenth-Century Miller*. Johns Hopkins University Press, Baltimore.

7. Iu. M. Lotman and B. A. Uspenskii. 1987. *Pis'ma russkogo puteshestvennika* Karamzina i ikh mesto v razvitii russkoi kul'tury. Introductory note to *Pis'ma russkogo puteshestvennika* by N. M. Karamzin. Nauka, Leningrad, 525–606.

was predicated on other mediation mechanisms such as acquisition of scientific credit by individuals, accumulation of institutional authority, and disciplinary legitimacy exchange.[8] According to this logic, the chapter is organized in three parts. The first part outlines the less familiar Soviet context of Ershov's career and explains the mechanisms of international community-building already in place by 1965 and shaping his visit to the U.S. The second part is a close study of contacts between the Akademgorodok Computer Center and Stanford University. The last part surveys the strategies of communication with the authorities and media, which reveal the experts' active role in promoting their techno-political visions.

How did knowledge production in a new field such as computing achieve its status as a field of international recognition? What was the interplay between the national and international dimensions of an emerging discipline? How did disciplinary, institutional, and personal agendas intersect in scientists' role as cultural diplomats? Approaching the question of how computers were transformed from a tool for computation into an object of scientific inquiry in a transnational perspective, this chapter emphasizes connections between two of this book's themes, "Defining the Discipline" and "Broadening the Profession." It engages a dialogue between several branches of computing history and the history of Cold War technoscience. Michael Mahoney studied how computing acquired a theoretical body of knowledge and traced the emergence of a coherent disciplinary identity during the 1960s as a consensus on an intellectual agenda combining automata theory, formal languages, computational complexity, and formal semantics.[9] William Aspray examined the institutional dynamic in the consolidation of computer science, predicated on crossing traditional disciplinary boundaries and negotiating with existing academic units.[10] Nathan Ensmenger outlined the new discipline's struggles for legitimacy and a checkered relationship with the growing computer industry, despite its apparent success in attracting students and funding.[11] Locating these American

8. J. Krige. 2006. Atoms for Peace, scientific internationalism, and scientific intelligence. *Osiris*, 21: 161–181.

9. M. Mahoney. 1997. Computer science: The search for a mathematical theory. In J. Krige and D. Pestre, editors. *Science in the Twentieth Century*, pp. 617–634. Harwood Academic Publishers, Amsterdam.

10. W. Aspray. 2000. Was early entry a competitive advantage? US universities that entered computing in the 1940s. *IEEE Ann. Hist. Comput.*, 22(3): 42–87.

11. N. L. Ensmenger. 2012. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*. MIT Press, Cambridge, MA.

processes not only as part of the academic-military-industrial complex but also in the context of transnational circulation of technology, people, and ideas, we gain a fuller understanding of computing practices and narratives during the Cold War.[12]

Beyond its distinctive Cold War aspect, the particular story of Ershov's trip to the U.S. also belongs to a more general study of computing communities in this volume, and it fits into a body of scholarship exploring the interaction between human and machine. The computing technologies were not, and still are not, generating numbers only for numbers' sake. The "computer revolution" and the ongoing proliferation of information technologies was accompanied by distinct ideas about science, the human mind, and social organization. While celebrated as technophiles and inventors, computer experts were often preoccupied with social and political concerns within their own milieu and on an imaginary "universal" scale.[13] From this point of view, transnational mediation is not an exceptional phenomenon born of Cold War conjuncture but a key site for inquiry into the nature of computer-related expertise. Professional organizations such as ACM emerge as key loci for studying the mechanisms of mediation. At the same time, for a full contextualization of the ACM's activities, it is also vital to account for the international dimension of experts' professional visions and networks.

9.1 Novosibirsk—Moscow—New York—San Francisco—Los Angeles

The itinerary of Ershov's trip to the U.S. in May-June 1965 was a snapshot of the landscape of Cold War computing. To begin, Ershov had to travel to Moscow from Novosibirsk, the location of his home institution, the Akademgorodok Computer Center. This part of his journey is a reminder that many Soviet computer-related developments took place beyond the capital, and rivalry often defined the relationships between different institutions. In Moscow, Ershov waited for travel permission from the U.S. Department of State. In the end, the trip took place about a week later than planned. Ershov was still able to attend the IFIP Congress but missed

12. For the politics of computing expertise and discourse in the United States, see R. Slayton. 2013. *Arguments That Count: Physics, Computing and Missile Defense, 1949–2012*. MIT Press, Cambridge, MA. For the international diffusion of computing, see J. Cortada. 2012. *The Digital Flood: The Diffusion of Information Technology across the U.S., Europe, and Asia*. Oxford University Press, New York. Also see J. R. Yost. 2008. *IEEE Ann. Hist. Comput.* 30(1): 2–3.

13. See B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals*. Association for Computing Machinery and Morgan & Claypool, New York.

the Princeton meeting of the IFIP working group on Algol and the session of the IFIP Technical Committee 2, where he was to represent the Soviet Union. Moreover, once in the U.S., the post-Congress part of the visit was not guaranteed—a visa extension had to be obtained, and the itinerary renegotiated.[14]

On one hand, the dense program of Ershov's post-congress trip was a reflection of the intensive growth in American computing. From the mid-1950s to the early 1960s, the field grew tremendously in both personnel and machinery, conquered foreign markets, and assumed a leadership position in the new discipline of computer science. After the congress, the Soviet expert flew to San Francisco. His program included a formal visit to Stanford, as well as informal interactions with local computer experts. Arriving in Los Angeles on June 2, Ershov visited Caltech and Allen Babcock Computing before his lecture to the ACM chapter. Ershov spent the next day at RAND, lecturing and visiting its Computer Center. He was back in New York on June 4 and had time to lecture at the Computer Usage company before flying to Moscow in the evening of the same day. In a week, Ershov traversed a range of important locations, from Stanford's newly created computer science department, to the ACM chapter, to the famous think-tank headquarters, to software firms, to private homes of the leaders of the community.[15]

On the other hand, it is Ershov's ability to enter each of these spaces that requires explanation—his invitations were initiatives from below—and by the same token reveals the mechanisms sustaining the process of international community building. Ershov's 1965 trip grew out of a pre-existing network of intellectual connections and personal contacts established during the decade preceding his trip and relying on the traditional mechanisms of scientific communication, such as publication exchange, and new international forums, such as IFIP. As a matter of fact, Ershov's trip had at least two pre-histories. One was related directly to the preparation for the 1965 IFIP Congress and started in June 1963, when Ershov became one of the organizers of the section on programming. The wider context leading to this moment was much broader and encompassed long-term circulation of computer-related knowledge across both the Atlantic Ocean and the Iron Curtain.

Already by the fall of 1955, when the existence of Soviet computers was officially announced in Darmstadt, West Germany, at the Conference on Electronic Digital Computers and Information Processing—the event that gathered European, Amer-

14. EA: Tim Standish to Andrei Ershov. May 28, 1965. f. 32/ l. 126–127.

15. EA: Introduction in Otchet "SShA-65." January 20, 1966. f. 46/ l. 1–13.

ican, and Soviet computer pioneers for the first time—Ershov already belonged to a small group of experienced programmers.[16] Among the first graduates of a new specialization in computational mathematics at Moscow State University (MGU), he was immediately employed at the first Soviet academic computer center, created during the same year under the leadership of academician A. A. Dorodnitsyn. There, he was working on the early Soviet automatic programming systems, the endeavor summarized in a pioneering monograph on the topic translated in English in 1959.[17]

If the Russian language name for automatic translators, "programming programs," pointed to the practical needs of automation of programmers' labor, it is "mathematical supply," the Russian term for "software," that revealed the peculiar national context shaping new practices of programming.[18] This mathematical view of programs was a result of the rise of Soviet cybernetics. Ershov explored the mathematical nature of programs as a graduate student under Alexei Liapunov. Famous as the patriarch of Soviet cybernetics, Liapunov was also the author of the program description method developed in 1953 and known as "programming schemata," an intellectual antecedent of the so-called higher-level programming languages.[19] A mentor to the core group of early Soviet programmers including Ershov, Liapunov and his students infused Soviet programming with mathematical abstractions and cybernetic visions of an empowering dialogue between humans and machines.

The geopolitical reconfigurations that followed the 1955 Geneva summit facilitated Soviet participation in the Darmstadt conference and, more famously, two high-profile projects of international science of the late 1950s: the Atoms for Peace program and the International Geophysical Year (IGY). The launch of Sputnik in 1957 prompted a surge of international interest in Soviet science and technology

16. On Soviet participation in the Darmstadt Conference, see S. Gerovitch. 2002. *From Newspeak to Cyberspeak: A History of Soviet Cybernetics*, p. 157. MIT Press, Cambridge, MA.

17. A. P. Ershov. 1959. *Programming Programme for the BESM Computer*. Pergamon Press, London.

18. For a brief overview of Western programming, see M. Mahoney. 2002. Software: The self-programming machine. In A. Akera and F. Nebeker, editors. *From 0 to 1: An Authoritative History of Modern Computing*, pp. 91–100. Oxford University Press, New York.

19. A. A. Liapunov. 1958. O logicheskikh skhemakh program. *Problemy Kibernetiki*, 1: 46–74. Fizmatgiz, Moscow. For a discussion of the method, see R. I. Podlovchenko. 2001. A.A. Lyapunov and A.P. Ershov in the theory of program schemes and the development of its logic concepts. In D. Bjørner, M. Broy, and A. V. Zamulin, editors. *Perspectives of System Informatics: 4th International Andrei Ershov Memorial Conference, PSI 2001, Revised Papers*, pp. 8–23. Springer, Berlin/Heidelberg.

and in Soviet computing in particular, as the Western observers speculated regarding the calculation means behind tracing the satellite's trajectory in the skies. The first bilateral exchange between Soviet and American computer specialists took place in 1958. Significantly, unlike the top-down Atoms for Peace or the IGY, which relied on an established tradition and infrastructure of scientific cooperation, this exchange was result of a bottom-up initiative.

John W. Carr, a specialist in numerical analyses, tried to reignite the contacts made in 1955 and to invite the Soviet experts to participate in a Michigan summer school of 1957, but with no results. By 1958 his arguments that keeping up with the Soviet computing developments was of strategic importance to national security got more traction, and the trips based on the principle of reciprocity were included under the umbrella of the bilateral agreement on exchanges, also known as the Lacy-Zarubin agreement, signed in January 1958.[20] Carr, by that time the president of the ACM, led a group of four experts to the Soviet Union in September 1958. Ershov was among the Soviet interlocutors of the Americans during their visit to Moscow, and took part in a long meeting with Carr and Alan Perlis that would prove to have long consequences.[21]

Ten years senior to Ershov, Perlis was among the MIT graduate students initiated into programming via Project Whirlwind, and already the author of the "mathematical language compiler" better known as IT language, as well as the leader of the ACM representatives involved in defining the International Algebraic Language, later renamed Algol.[22] Perlis was particularly well placed to introduce Ershov to the project of defining a universal algorithmic computer language—a language capable of transcending the characteristics of particular machines and as a consequence also allowing the exchange of such machine-independent programs. Ershov was recruited to the universal language agenda on the spot.[23] While his eagerness was a spontaneous reaction to a stimulating communication with the American specialists, his training under Liapunov and the grandiose visions of Soviet cybernetics provided a rich soil for the Algol project's ambitions.

---

20. Y. Richmond. 2004. *Cultural Exchange and the Cold War: Raising the Iron Curtain*, p. 15. Penn State University Press, University Park.

21. For more on the first Soviet-American exchanges in computing see K. Tatarchenko. 2013. 'A House with the Window to the West': The Akademgorodok Computer Center, 1958–1993. Ph.D. dissertation, Princeton University, Chapter 3.

22. D. Nofre. Alan J. Perlis. Available at http://amturing.acm.org/award_winners/perlis_0132439.cfm.

23. EA: Ershov Diary entry for September 4–7, 1958. f. 35/ l. 101–102.

The complexity of Algol history is known thanks to the growing literature on the topic.[24] Following the first discussions about a machine-independent international language at Darmstadt, the German-Swiss-American group gathered in Zurich in summer 1958 and produced the text known as the Algol 1958 report. The publication of the report in *Communications of the ACM* and several other outlets generated interest in further discussions and proposals for improvement.[25] More conferences and the Algol 60 report followed, producing international momentum around the language in which the Soviet experts partook. In the aftermath of the American visit, Ershov continued correspondence and literature exchange with Carr and Perlis, and also benefited from their international networks.

In March 1960, Peter Naur, the editor of the *Algol Bulletin* that became the main forum for the Algol group discussions, responded to Ershov's questions and expressions of gratitude for receiving the *Algol Bulletin*: "Yes, it is indeed Prof. Perlis who during the last Paris meeting gave me your name and address . . . " The Danish computer pioneer was eager to correspond in Russian, "I am always interested in keeping the little I know in use," explained Naur. He would also be glad to send the *Algol Bulletin* to other interested Soviet experts. Yet he could provide little information to reassure the Soviet programmer regarding the institutional status of the Algol group, which had no fixed statutes or official membership structure.[26] This situation was soon to change.

In parallel to the Algol project, another transnational initiative led to the International Conference on Information Processing under the auspices of UNESCO. Held in the Paris headquarters of UNESCO in June 1959, the conference led to the establishment of a new non-governmental professional body for computer experts, the International Federation for Information Processing (IFIP). Structured as a society of societies, the Federation's goal was to foster international cooperation and public recognition of the new field. With the Soviet Academy of Sciences among its founding members, the IFIP and its triennial congresses facilitated encounters between the East and the West.[27]

24. See the special issue devoted to Algol history: H. Durnova. 2014. Embracing the Algol effort in Czechoslovakia. *IEEE Ann. Hist. Comput.* 36(4): 26–37.

25. A. J. Perlis and K. Samelson. 1958. Preliminary report: International algebraic language. *Commun. ACM*, 1(12): 8–22.

26. EA: A. Ershov to P. Naur. February 22, 1960. f. 124/ l. 206–208; P. Naur to Ershov. March 7, 1960. f. 135/ l. 289.

27. K. Tatarchenko. 2010. Cold War origins of the International Federation for Information Processing. *IEEE Ann. Hist. Comput.* 32(2): 46–57.

The content of the IFIP's activities, organized in technical committees and working groups, accommodated many facets of computing practices and became an attractive structural umbrella for the growing responsibilities of the Algol group. The Algol group members had formally transmitted the responsibility for the development, refinement, and specification of Algol to the IFIP Working Group 2.1, officially created in 1962 as a "home for Algol."[28] Facing the difficulties of implementation as well as proliferation of corrections, subsets, and extensions in the early 1960s, the group moderated some of its early absolutism regarding the definition of an ultimate language. Instead, Algol became a model for research in programming. The Algol group meetings, formally devoted to the definition of new versions of Algol, became occasions for debating and imagining the future of programming. The official aim of the Working Group 2.1 is still "to explore and evaluate new ideas in the field of programming, possibly leading to the design of new languages."[29]

On the national level, the Soviet participation in IFIP and Algol activities was secured thanks to the efforts of Dorodnitsyn, the head of the Computer Center of the Soviet Academy of Sciences and Ershov's patron. Addressing the academic hierarchy, Dorodnitsyn couched his arguments in the political terms of national prestige, scientific intelligence, and peacemaking through scientific collaboration.[30] The resulting circulation of people, ideas, and artifacts led to the most unexpected results. Recognized for its importance in teaching and as a publication standard, Algol was often considered as a practical failure in the U.S., where IBM's Fortran continued to dominate scientific computation, and a moderate success in Europe. At the same time, the diffusion of several Algol compilers in the East, made Algol the most popular programming language in the Soviet computer centers of the 1960s.[31]

Ershov was at the very heart of the Soviet Algol efforts in two capacities. He was the mastermind behind the diffusion of information on Algol, as translator and editor of Algol 58 and Algol 60 reports.[32] Moreover, Ershov had a project to

---

28. *Algol Bulletin* 15.4. The IFIP Working Group on ALGOL, p. 52. Available at http://archive.computerhistory.org/resources/text/algol/algol_bulletin/A15/P4.HTM (accessed January 2016).

29. See the description available at http://foswiki.cs.uu.nl/foswiki/IFIP21/ and IFIP website available at http://www.ifip.org/bulletin/bulltcs/tc2_aim.htm (accessed January 2016).

30. EA: A. A. Dorodnitsyn to A. V. Topchiev. December 15, 1959. f. 24/ l. 5–6, 9–10.

31. For Algol in Eastern Europe, see Durnova, Embracing the Algol effort in Czechoslovakia.

32. A. J. Perlis and K. Samelson. 1959. *Soobshchenie ob algoritmicheskom iazyke ALGOL*, translated by A. P. Ershov. Vychislitel'nyi Tsentr AN SSSR, Moscow); A. P. Ershov. 1960. Predislovie redaktora perevoda. In P. Naur, editor. *Soobshchenie ob algoritmicheskom iazyke ALGOL 60*, pp. 3–9.

implement a subset of Algol and circulated his group's definition of the "Input language" via the *Algol Bulletin*.[33] A letter from Carr indicates the positive reception of Ershov's work in the West. "I am very pleased that your group has done so much with Algol-60," wrote Carr to Ershov. "I would suspect that you may have gone further, faster than any other group in the world, except for the group under Dijkstra at Amsterdam."[34] Mike Woodger's invitation for the IFIP Congress programming section was another endorsement of Ershov and his group's work on Algol.[35] At the congress and elsewhere during the American trip, Ershov delivered a description of the automatic programming system implemented under his leadership in Siberia, where the young expert became the head of the programming department first at the Institute of Mathematics and later at the Computer Center.[36]

The 1955 Darmstadt conference started the process of international community-building. In his pioneering study of international computing, William Aspray observed that the scientific forms of communication had prevailed in post-war American-British computing, and Atsushi Akera pointed to the connection between forms of scientific communication and accumulation of personal authority and credit, most notably by John von Neumann.[37] Akera also argued that the identity of the new field of computing was constructed on the intersection of multiple disciplines and different institutions, a process that relied on the mechanism of "dissociation," namely the letting go of attachments to intellectual agendas and institutional allegiances.[38] The international community-building through the late

Vychislitel'nyi Tsentr AN SSSR, Moscow/Novosibirsk, 3–9; A. P. Ershov, S. S. Lavrov, and M. R. Shura-Bura. 1965. In P. Naur, editor. *Algoritmicheskii iazyk ALGOL 60: Peresmotrennoe soobshchenie*. Mir, Moscow.

33. A. P. Ershov, G.I. Kozhukhin, and U. M. Voloshin. The input language system of automatic programming. *Algol Bulletin Supplement*, 14. Available at http://archive.computerhistory.org/resources/text/algol/ACM_Algol_bulletin/1064063/frontmatter.pdf (assessed January 2016). Also see A. P. Ershov. 1963. *Input Language for Automatic Programming Systems*. Academic Press, New York.

34. EA: J. W. Carr to A. Ershov. November 10, 1961. f. 135/ l. 192. For a comparison of German and Dutch compilers for Algol, see G. Alberts and E. G. Daylight. 2014. Universality versus locality: The Amsterdam style of Algol implementation. *IEEE Ann. Hist. Comput.* 36(4): 52–63.

35. EA: M. Woodger to A. Ershov. June 21, 1963. f. 134/ l. 148.

36. A. P. Ershov. 1965. The alpha—An automatic programming system of high efficiency. *Proceedings of IFIP Congress 65*, volume 2: 622–623. Macmillan and Co., London.

37. W. Aspray. 1986. International diffusion of computer technology, 1945–1955. *IEEE Ann. Hist. Comput.* 8(4): 351–360.

38. A. Akera. 2008. *Calculating a Natural World: Scientists, Engineers, and Computers during the Rise of US Cold War Research*, p. 21. MIT Press, Cambridge, MA.

1950s and early 1960s represented the next step—the establishment of shared spaces of encounter. Such forums included working groups of national professional societies, such as ACM and Society of Applied Mathematics and Mechanics (Gesellschaft für Angewandte Mathematik und Mechanik or GAMM), international non-governmental bodies such as IFIP, and less visible epistolary networks sustaining intellectual coordination and personal contacts in between meetings. What was exchanged and acquired via these transnational forums and networks? Scientific information—algorithms and formulas, language descriptions and corrections, news on hardware and plans on implementations—but also intangible entities. Personal trust and sympathy, scientific credit, and institutional authority were not immediate but long lasting products of international encounters.

9.2 Person-to-Person, Institution-to-Institution, Discipline-to-Discipline

The primary reason for Ershov's American visit, the IFIP Congress in New York, was the biggest gathering of international computer experts yet. About 5,000 attended the main scientific program, and many more went to the exhibit. In conference halls and on the exhibition floor, the biggest excitement at the congress was time-sharing. The illusion of instantaneous interaction with a machine, despite physical distance and multiple users, put this new technology at the central place of an imagined social-technological order, where all problems of mankind could be solved by the computer, the ultimate human helper. For example, participants discussed the possibility of setting up an international transatlantic data-link under IFIP support, a project illustrating the momentum of general enthusiasm around the computer as a tool of internationalism. Such discussions and projects also witnessed the change operating in the intellectual geography of computer-related knowledge: cybernetic vocabulary and a broader research agenda encompassing man-machine interaction were incorporated under the umbrella of information technology and its multiplying applications. The journalists covering the event did not shy from proclaiming the symbolic triumph of computing over cybernetics: "It was nice. . . . It was grand. . . . We got a new perspective on the potent importance of computers to the world. . . . Across the street an Institute for Cybercultural Research held its conference. They said 'Doom is coming'."[39]

39. EA: *Computing Newsline*, vol. 2, no. 11, June 2, 1965, f. 32/ l. 68–75. The Institute of Cybercultural Research also held its membership meeting on May 29, 1965. In 1966, the institute organized a pioneering conference including such prominent speakers as Hannah Arendt, see H. Arendt.

The differences in the disciplinary fates of cybernetics and computer science offer abundant insights. Geof Bowker analyzed the rise of cybernetics as a new universal science in the 1950s as a result of particular rhetorical strategy—cybernetics provided a site for "legitimacy exchange" between several disciplines.[40] Ron Kline argued that this strategy backfired when multiple meanings of cybernetics proliferated, and the field became associated with the fantastical claims of the so-called fringe groups, the "wrong" partners in the exchange game. In addition, the difficulties of institutionalizing American cybernetics in the early 1960s belonged to a particular Cold War context: Soviet cybernetics was on the rise in the service of communism, a development that simultaneously made cybernetics intellectually suspicious and the object of state patronage. The CIA became directly involved in sustaining professional activities, such as the establishment of the American Association for Cybernetics in the mid-1960s, but could not prevent the general decline in the field.[41]

Cybernetics' loss was a gain for the up-and-coming discipline of computer science. It too was directly benefiting from the largesse of state and military funding and the growth of the military-industrial complex. It too was a "synthetic" discipline of contested legitimacy and in the quest for identity. It too made questions of interaction between man and machine one of its key research areas but came to emphasize a distinct set of strategies. Computer science drew on the universality, international authority, and administrative power of an already established discipline, that of mathematics.[42] Unlike the "cybernetic gap" that strengthened the politicization of cybernetics, computer science was constructing an image of itself as simultaneously strategic and "apolitical." The research on formal languages or the theory of algorithms raised similar issues everywhere and was predicated on the international flow of knowledge.

1966. On the human condition. In A. M. Hilton and Institute for Cybercultural Research, editors. *The Evolving Society: The Proceedings of the First Annual Conference on the Cybercultural Revolution—Cybernetics and Automation*, pp. 213–220. Institute for Cybercultural Research, New York.

40. G. Bowker. 1993. How to be universal: Some cybernetic strategies, 1943–70. *Soc. Stud. Sci.*, 23(1): 107–127.

41. R. R. Kline. 2015. *The Cybernetics Moment: Or Why We Call Our Age the Information Age*, chapter 7. Johns Hopkins University Press, Baltimore.

42. M. Mahoney. 1997. Computer science: The search for a mathematical theory. In J. Krige and D. Pestre, editors. *Science in the Twentieth Century*, pp. 617–634. Harwood Academic Publishers, Amsterdam; and M. Tedre. 2014. *The Science of Computing: Shaping a Discipline*, Chapter 3. Chapman and Hall/CRC, London/New York.

That Ershov received the California invitation and was able to spend a whole day visiting Stanford's new department of computer science was part and parcel of discipline-building. This process involved the acquisition of international authority. As for cybernetics, the Cold War and the rise of Soviet cybernetics were an inevitable part of the context, but in computer science their influence played out in a different way. Individual contacts, conflicts and interests, as well as institutional arrangements and patronage, were responsible for these crucial differences.

In 1965, Ed Feigenbaum, the future Turing award-winner now known as the father of expert systems, was on the move.[43] Geographically, he was not moving too far: from Berkeley to Palo Alto. But speaking of professional identity, this was a significant step: from the University of California Berkeley, School of Business and courses on organization theory, to Stanford University, its new Computer Science department and teaching Algol; from "how humans think, which occupied much of his time at Berkeley, to the technology of getting computers to think."[44] Before joining Stanford in the fall 1964, Feigenbaum was able to get some time off teaching and traveled in Europe and the Soviet Union, where he was invited to lecture on artificial intelligence and heuristic programming.

For the members of the Cybernetic Council of the Soviet Academy of Sciences, who were organizing the American expert's trip in October 1964, there were no doubt about his disciplinary belonging—a pupil of Herbert Simon, Feigenbaum was a cybernetician. In August 1965, when Feigenbaum addressed the audience at the ACM national conference with a talk about his trip, the title was "Soviet Computer Science, Revisited."[45] The mismatch between the Soviet and the American mutual labeling was indicative of two things. One was the affirmation of his disciplinary allegiance on the part of Feigenbaum. His travels to the Soviet Union translated into an invited talk to one of the most prestigious national forums thanks to the suggestion of George Forsythe, the ACM president and the head of Stanford's Computer Science department.[46] Another was the diverging national disciplinary trajectories in the Soviet Union and the United States.

43. See J. November. 2012. *Biomedical Computing: Digitizing Life in the United States*, Chapter 5. Johns Hopkins University Press, Baltimore.

44. N. J. Nilsson. Ed Feigenbaum. Available at http://amturing.acm.org/award_winners/feigenbaum_4167235.cfm.

45. Feigenbaum Papers (hereafter **FP**): E. Feigenbaum. 1965. Soviet computer science revisited. Box (b.) 50, Folder (f.) 36, druid:dq750wz7335. The papers are at the Stanford University Libraries, Department of Special Collections and University Archives.

46. FP: S. Rosen to E. Feigenbaum. February 16, 1965. b. 26, f. 1, id: druid:jj004gp9798.

The epithet “revisited” indicated that this was not the first trip to the Soviet Union for Feigenbaum. In fact, he was among the attendees of the 1960 International Federation of Automatic Control Congress in Moscow. This gathering became notorious for the Soviet enthusiasm when receiving Norbert Weiner among the congress guest speakers. As a graduate student Feigenbaum was curious about the Soviet Union and eager to travel but had little money; his adviser had many contacts with RAND, which had the money and was also interested in information about the congress. Feigenbaum’s report to RAND indicated that the congress had little to offer in terms of scientific exchange but that person-to-person communication with the Soviets was “mutually enlightening.” Thanks to contacts from the previous American experts’ trips, Feigenbaum was able to deviate from the official program of the congress and meet with Ershov.[47]

At that time, Ershov was already officially the head of the programming unit at the new Institute of Mathematics created at Akademgorodok, but was still working in both Moscow and Novosibirsk. The personal interaction between the two young researchers went well. “Dr. Ershov,” wrote Feigenbaum in the report, “is a man of great intelligence, curiosity, and energy, and he has considerable awareness of the state of the computing art in Russia as well as in the United States.”[48] They maintained an active correspondence. When Feigenbaum informed Ershov of his 1964 visit to the USSR, Ershov secured the help of the head of the Siberian Branch of the Academy of Sciences, mathematician M. A. Lavrentiev, in order to add Novosibirsk to the American’s itinerary.[49] Summarizing his impression from his 1964 visit, Feigenbaum praised the “splendid week” that he spent in Siberia and described Akademgorodok as a “fascinating place.”[50]

He also commented on the new institution that acted as his host. “The Computer Center,” observed the American expert, “is the best I have seen on my two trips (in terms of the ideas that the American computer scientist would consider sound, modern, and forward-looking).”[51] In his private letters to Ershov, Feigenbaum was no less explicit and eloquent:

47. EA: E. Feigenbaum to A. Ershov. June 14, 1960. f. 135/ l. 274–275.

48. FP: Memorandum RM-2799. 1961. b. 30, f. 35, id: druid:tk542jp7648, on p. 10. See also E. A. Feigenbaum. 1961. Soviet cybernetics and computer sciences, 1960. *Commun. ACM*, 4(12): 566–579.

49. EA: M. A. Lavrentiev to N. M. Sisakian. August 8, 1965. f. 37/ l. 150–151.

50. About 20 letters exchanged between Ershov and Feigenbaum from 1960–1964 are preserved in the Ershov Archive.

51. FP: E. Feigenbaum. 1965. Soviet computer science revisited. b. 50, f. 36, druid:dq750wz7335.

> . . . nowhere else did my scientific discussions take place at such a high level, and nowhere else were they as "equal" (you will know what I mean). Also nowhere else in the Soviet Union did I find such a complete understanding of the American computer scene. It is really true that you are the Novosibirsk branch of the ACM.[52]

Even if exaggerated, Feigenbaum's description captured the particular position of Ershov and his department regarding their disciplinary affiliation.

In fact, even as an author of a monograph on automatic programming—a "magnum opus" according to Feigenbaum—Ershov was not yet a "doctor" when the young American first met him in 1960. His degree was awarded in 1962, when his adviser, Liapunov, also moved to Akademgorodok and was able to secure enough supporters among the mathematicians to proceed with the defense—the difficulties pointing to struggles in the Soviet scientific community regarding the status of programmers' work. The occasion was not very festive, however. Liapunov and Ershov already felt divided regarding the direction of Ershov's research, the conflict over which nearly destroyed Ershov's programming group and its work on implementing Algol, the Alpha system. When the Computer Center was separated from the Institute of Mathematics and became an independent institution, Ershov's group moved out, creating an institutional rupture and making clear the diverging intellectual agendas between cybernetics and programming.

Feigenbaum's visit to Akademgorodok as the guest of the new Computer Center was mentioned in the local press and highlighted the international authority and public visibility of the new institute. Summing up the results of Feigenbaum's visit, his hosts, Ershov, and the director of the Computer Center, Gurii Marchuk, briefly outlined scientific intelligence gathered on heuristic programming, time-sharing, and new hardware. But instead of information collection, the two hosts stressed the productivity of scientific interactions. They concluded that transmitting information about Soviet developments promoted the prestige of Soviet science, the Siberian Branch of the Academy of Sciences, and that of the Computer Center. The Siberian scientists explicitly named another, additional benefit of the visit—the potential of cooperation between institutions. Feigenbaum's visit connected the Computer Center with Stanford University, "which like MIT is becoming one of the most interesting academic centers for computational sciences."[53] Feigenbaum enthusiastically carried out that potential reciprocating Ershov's hospitality at home in the San Francisco Bay Area.

52. EA: E. Feigenbaum to A. Ershov. December 22, 1964. f. 133/l. 174–175.

53. EA: A. P. Ershov, and G. I. Marchuk. December 11, 1965. Otchet o visite. f. 62/ l. 88–92.

Ershov visited Stanford on June 1, 1965. He spent most of the morning in informal discussions with programmers in Polya Hall. Next, he talked and had a long lunch with Niklaus Wirth, the assistant professor of computer science and also a member of the Algol group. The afternoon program included Ershov's own talk and more conversations, coffee breaks, and visits with philosopher Patrick Suppes and psychiatrist Kenneth Colby. A photograph shows Ershov with John McCarthy exploring the time-sharing terminal. The day ended with a dinner party at the Forsythes.[54]

The list of guests preserved in Ershov's trip documents encompasses the history of American computing on a page.[55] The guests included people of different generations. Harry Huskey worked on ENIAC, designed the NBS SWAC computer in Los Angeles, and was Niklaus Wirth's supervisor at Berkeley. John "Jack" Herriot was the mathematician working in numerical analyses who served as the first director of the Stanford Computer Center in the 1950s and supported Forsythe in establishing the new department. Gene Amdahl was the chief architect of IBM System/360 announced with great pomp about a year prior to the party. Cleve Moler was doing graduate work under George Forsythe, and his fame as the creator of Matlab was yet to come.[56] Four among the guests were to receive the ACM Turing award: Stanford's own John McCarthy (1971), Niklaus Wirth (1984), and Edward Feigenbaum (1994), as well as Marvin Minsky (1969), visiting from MIT.[57] Other Stanford faculty would become prominent members of the American academic-military-industrial complex: William Miller from Argonne National Lab would become Stanford's vice-provost; Richard Watson would work for Shell Oil and Lawrence Livermore; Gene H. Golub became the SIAM president. Another visitor from MIT, Seymour Papert, the future co-inventor of Logo programming language, would exercise a particularly strong influence on the later career of Ershov and his preoccupation with computer literacy.

The party and the visit more generally was a great success. The American hosts were delighted by Ershov, both personally and professionally. "Congratulations splendid Californian visits," telegraphed Edward Feigenbaum on June 19th to Novosibirsk, "eagerly waiting confirmation safe return home all requests

54. EA: Schedule. June 1, 1965. f. 32 / l. 140.

55. EA: Guests at Forsythes' House. June 1, 1965. f. 32/ 157.

56. T. Haigh. 2008. Cleve Moler: Mathematical software pioneer and creator of Matlab. *IEEE Ann. Hist. Comput.*, 30(1): 87–91.

57. See the list of ACM Turing award winners. Available at http://amturing.acm.org/alphabetical.cfm.

already satisfied warmest regards."[58] A tangible expression of appreciation was the sheer amount of technical information and scientific literature shared with the Soviet specialist—the parcels kept arriving for months, containing tens of books, blueprints and preprints. The "satisfied requests" mentioned by Feigenbaum were no doubt related to literature acquisition. But the best proof of the mutual appreciation was that Ershov was re-invited, this time for a long-term visit. "Ed Feigenbaum, John Herriot, John McCarthy, William Miller, Niklaus Wirth, myself and others," wrote George Forsythe in September 1965, "are very much impressed with the quality of your research and the breadth of your ideas. We hope ever so much that you and your family will be able to come here. . . . I cannot stress too highly how anxious we are for you to visit us."[59]

Although Ershov would not obtain the permission for such a trip and would return to Stanford only five years later, the cooperation between the institutions developed thanks to the collaboration between Ershov and John McCarthy and the latter's frequent visits to Akademgorodok in the 1960s.[60] In the 1970 request to support Ershov's application for an American visa, Forsythe employed the official rhetoric of reciprocity backed up by somewhat inaccurate statistics. The list included two visits by Feigenbaum (1960, 1964), five visits by John McCarthy (1965, 1966, 1967, 1968, 1970) and his own visit in 1967 summing up to the total of about five months vs. Ershov's two-day visit in June 1965 and Marchuk's three-day visit in December 1967. Forsythe's aim was to stress the imbalance in the contacts between two institutions. Commenting on Ershov's friendliness to Americans and the ease of communication, Forsythe concluded that "it is very much in the national interest for Yershov [sic] to be granted a visa to come here."[61] The nature of the interest was not specified.

Forsythe was persistent in his support for cooperation with the Akademgorodok Computer Center and Ershov in particular. Part of the explanation was the appeal of Ershov's professional profile—a theoretical thinker, a system builder, a manager,

58. EA: E. Feigenbaum to A. Ershov. June 19, 1965. f. 133/ l. 93.

59. EA: G. Forsythe to A. Ershov. September 27, 1965. f. 133/ l. 50.

60. For more on Ershov and McCarthy, see K. Tatarchenko. 2011. Informatika ot Silikonovoi Doliny do Zolotoi Doliny. In A. N. Tomilin, editor. *SoRuCom–2011 Proceeding, Second International Conference on the History of Computers and Informatics in the Soviet Union and Russian Federation*, pp. 278–282. Veliky Novgorod.

61. FP: G. Forsythe to A. Wortzel. Soviet and East European Exchange Staff, Department of State. February 19, 1970. b. 10, f. 4, id: druid: zg821bk7327. The copy went to two other officials but a blind copy also included P. Armer, J. McCarthy, E. Feigenbaum, and W. Ware.

and a teacher. Forsythe's publications where he articulated the relation between theory and practice in computer science in comparison to mathematics and physics show how such a profile corresponded to his vision:

> . . . to a modern mathematician, design seems to be a second-rate intellectual activity. But in the most mathematical of the sciences, physics, the role of design is highly appreciated. . . . If experimental work can win half the laurels in physics, then good experimental work in computer science must be rated very high indeed.[62]

Forsythe's preoccupation with defining the values of the new discipline also transpired in his presidential letters to ACM members, such as the one printed in the March 1965 issue of the *Communications of the ACM*. His calls for a collective work on identity construction were not limited to the American constituency and relevant to the international community:

> We must now turn our attention from the battle for recognition to the struggle to recognize the identity of our new discipline. . . . The core of Computer Science has become and will remain a field of its own, concerned with the forefront of new ideas. . . . I conclude that the computer and information sciences badly need an association of people to study them, improve them, and render them better understood and thus more useful.[63]

If Forsythe saw Ershov and cooperation with the Akademgorodok Computer Center as an asset for consolidating the disciplinary identity of computer science, on his side Ershov was a receptive interlocutor on whom Forsythe's efforts at internationalization and identity-definition made a strong impression.

Ershov and his hosts were behaving as if they were in a neutral virtual space defined by shared problems and saw international cooperation as a tool to their solution. "I should repeat myself," wrote Ershov to George Forsythe in the aftermath of his visit, "saying how much impressed I am by your department—a strong collective of researchers, good care of students and most interesting plans—and most importantly, by a singular commonality of our problems and research interests."[64] In a few years, Ershov would climb the Soviet academic hierarchy and become a

62. George Forsythe cited in D. E. Knuth. 1972. George Forsythe and the development of computer science. *Commun. ACM*, 15(8): 721–727, on 723.

63. G. Forsythe. 1965. President's letter to the ACM membership: Why ACM? *Commun. ACM*, 8(3): 143–144.

64. EA: A. Ershov to G. Forsyth. June 28, 1965. f. 126/ l. 157.

tireless advocate for a Soviet disciplinary equivalent of the computer science, first under the banner of "theoretical programming" and later as "informatics."

In 1967, Ershov and Liapunov coauthored an article devoted to the definition of "program," which demarcated the intellectual territory between cybernetics and theoretical programming and proclaimed the latter a synthetic mathematical science drawing on theory of algorithms, logic, and graph theory.[65] Ershov concluded his talk at the IFIP 1971 congress held in Ljubljana, Yugoslavia, with a declaration of an international rivalry in the new field: "Scientific groups in Stanford, Berkeley, Cambridge (MIT), New York, Kiev, Novosibirsk and some other places, previously scattered, are now entering into a period of intensive scientific competition which will result in rapid changes in the theory of programming . . . "[66] He was self-consciously manipulating the discourse of discipline-building and international competition in both domestic and international realms. "Lately, I endeavor to create the theory of programming (our equivalent as an independent discipline). My presentation at the IFIP Congress is one of the results," explained Ershov to John McCarthy at the end of 1971.[67]

Friends and rivals, the members of the international computing community were well aware of academic hierarchies and the logic of defining one's access to funding and promotion. From internal consensus on the intellectual agenda to external recognition of legitimacy, building a new discipline was a complex task. It depended on national and institutional conjunctures, local conflicts, and international alliances. Both Stanford's Computer Science Department and Ershov's group at the Akademgorodok Computer Center simultaneously appropriated cybernetic elements into their agendas and drew a demarcation line between cybernetics and computer science. The new discipline of computer science possessed a mathematical theoretical core transcending all borders and promised applications of strategic importance to national security.

9.3 Twisted Truths: Dealing with Hazards of Boundary-Crossing

Ershov's meeting with the members of the ACM Southern California chapter occurred the very next evening after the party at Forsythes. Unlike in Stanford, no cutting-edge research was discussed. He delivered a general lecture on the state of

65. A. P. Ershov and A. A. Liapunov. 1967. O formalizatsii poniatiia programmy. *Kibernetika*, 5: 40–57.

66. EA: Ershov. Doklad na IFIP 1971. f. 434/ l. 192.

67. EA: A. Ershov to J. McCarthy. December 2, 1971. f. 84/ l. 218.

computing in Novosibirsk and answered questions from the public. In Los Angeles, as elsewhere during his trip, Ershov talked in front of a packed room. The pictures printed in the *ACM Data-link Bulletin* (the bulletin of the local chapter) showed smiles, hands gesticulating in engaged conversation, and drinks consumed in a festive mood. *The Computer Newsline*, a small newsletter based in Santa Monica, described the impressions of Ershov's talk in the most positive terms: "He's a winner. There wasn't a person in the audience that wouldn't have liked to know him. We even laugh at the same aspects of computing, i.e, he's written a routine called 'garbage collector.'"[68]

Ershov's visit to RAND similarly went well. Yet Ershov's presence on the grounds of the headquarters of America's most prominent think-tank—labeled "the Academy of Death and Destruction" by the Soviet media—provokes some puzzlement.[69] The visit appears less surprising when considered in the context of RAND's institutional attitude to information. RAND was not CIA and defended the position that the quality of intelligence gathered in open sources and by expert witnesses was at least equal to what could be accomplished by professional agents using espionage techniques. In particular, the 1959 visit of American computer experts to the Soviet Union included RAND's Willis Ware, as a member from the Institute of Radio Engineers (IRE), and Paul Armer, an ACM member. This visit led to a vast collection of Russian language publications on Soviet computing and cybernetics and gave RAND a head-start as the center of expertise on the topic.[70] One of the results was the establishment of *Soviet Cybernetics: Recent News Items*, later known as *Soviet Cybernetics Review*.[71] Although no published descriptions of Ershov's visit to RAND remain, pictures show the Soviet specialist talking in front of the backboard—the very symbol of mathematical abstraction and universality (Figure 9.1).

The scandal following the media coverage of Ershov's trip was a reminder of how fragile such relationships were despite the tremendous efforts of organizers.

68. EA: *Computing Newsline*, vol. 2, no. 11, June 2, 1965. f. 32/ l. 68–75.

69. W. H. Ware. 2008. *RAND and the Information Evolution: A History in Essays and Vignettes*, p. 163. RAND Corporation, Santa Monica.

70. W. H. Ware. Soviet computer technology: 1959. RAND, RM-2541. Available at http://www.rand.org/content/dam/rand/pubs/research_memoranda/2008/RM2541.pdf (accessed January 2016); see also W. H. Ware, Soviet computer technology—1959. *Commun. ACM*, 3(3): 131–166.

71. A. Bochannek. 2011. What did the Americans know? A review of select periodicals. In A. N. Tomilin, editor. *SoRuCom–2011 Proceeding, Second International Conference on the History of Computers and Informatics in the Soviet Union and Russian Federation*, pp. 3–7. Veliky Novgorod. For more on RAND and Soviet computing, see N. Lewis. 2016. Peering through the curtain: Soviet computing through the eyes of western experts. *IEEE Ann. Hist. Comput.*, 38(1): 34–47.

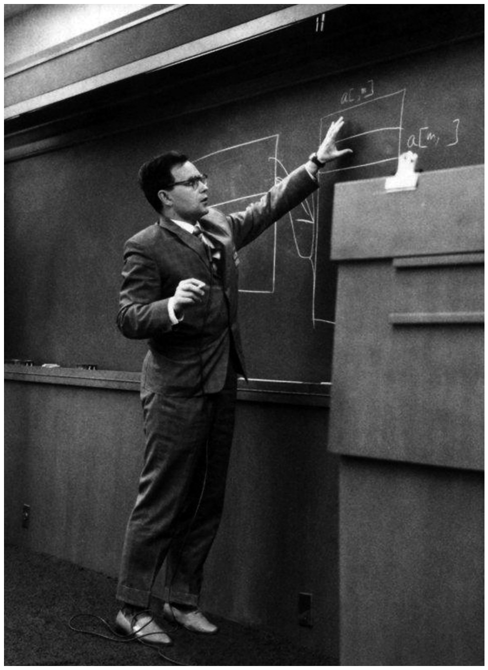

Figure 9.1 **Ershov lecturing at RAND (1965).** (Courtesy of © Photo archives of the Siberian Branch of the Russian Academy of Sciences)

The way that the protagonists managed the crisis provides additional insights into the nature of Cold War scientific communication and the importance of the elusive but critical factors of trust, responsibility, and personal agency. Nikolai Krementsov introduced the notion of "double loyalties" to describe international genetics in the interwar period.[72] This notion aptly captures the communication practices of Cold War computer scientists. Such double loyalties, one to the international scientific community and another to national structures and patrons, were not mutually exclusive. On the contrary, the upholding of both loyalties and associated networks was an essential feature for acquiring disciplinary authority and securing the material basis for one's research agenda. How were these loyalties enacted? Both Soviet and American specialists became agile at manipulating the rhetoric of internationalism and competition in official documents, private communications,

72. N. Krementsov. 2005. *International Science between the World Wars: The Case of Genetics*. Routledge, New York.

and public forums. Double loyalties took extra efforts and created multiplying narratives, a proliferation that explains the very richness of documentary traces.

Feigenbaum's correspondence shows his attention to those working behind the scenes to enable international encounters. On February 26, 1965, Feigenbaum wrote a thank you letter to the head academic secretary of the Soviet Academy of Sciences, N. M. Sisakian, enumerating the productive scientific discussions and specifying the moments of disorganization that impeded some of the interactions during his trip. The key idiom that the "community of scientists is truly universal, crossing national borders" made clear how much effort such crossing took on behalf of scientists as well as supervising bodies.[73] The same was true for organizing Ershov's trip in the U.S. On June 24, Feigenbaum addressed Franc Siscoe, the head of the Soviet and Eastern European Exchange Staff at the Department of State, with a formal letter of gratitude for the assistance with arrangements for Ershov's visit and the overall handling of the IFIP congress.[74] Such letters were not required by the rules; they demonstrate the social aspect of interactions between the state apparatus and the scientists.

Another exchange of letters also highlights that "the state" was not a monolithic structure and that personal horizontal networks of exchange connected people with different affiliations: access to information was an asset. In February 1965, Feigenbaum corresponded about his Soviet visit with an officer from the Air Force Office of Scientific Research. In fact, his 1964 trip was sponsored by RAND, but not coordinated with branches of the Air Force. As a result, his "circuit" overlapped with another visit by a young specialist in computing sponsored by the Air Force and traveling on a "tourist" visa. The exchange itself was promoted by the officer's informal request for Feigenbaum's report on the trip explained by a heavy personal commitment to "a lot of this field" and accompanied by an offer to share more reports. However, Feigenbaum was reluctant to fulfill this request. While promising to "evoke" a copy from RAND, he suggested his interlocutor use her official capacity. Feigenbaum explained that the draft of 137 pages was "rather tightly held by RAND" and a "sterilized" version was to appear for larger circulation.[75]

From the "universal science" crossing national borders to a "tightly held" report caught in the disjunction between bureaucratic branches, Feigenbaum's writings preserved contradictory images of the nature of international encounter. His choice

73. EA: E. Feigenbaum to N. M. Sisakian. February 26, 1965. f. 133/ l. 146–147.

74. FP: E. Feigenbaum to Franc Siscoe. June 24, 1965. b. 26, f. 25, id: druid:xg617zt2629.

75. FP: R. Swanson to E. Feigenbaum. February 8, 1965; E. Feigenbaum to R. Swanson. February 18, 1965. b. 26, f. 25, id: druid:wy197vn0686.

of words also holds clues to how experts managed such contradictions. Switching hats was a frequent metaphor for the Western scientists to describe their multiple professional obligations such as RAND and ACM affiliations for Paul Armer, or the academy and military work for Alan Perlis. The expression aptly caught the professional identity that could not fit narrow definitions and was predicated on a personal capacity to adjust to multitasking with ease. Beyond self-promotion, the experts were often driven by visions of future developments in computing. This visionary aspect, partially inherited from cybernetics, helps explain both their interactions with bureaucracy and with media.

The "hat off" was Feigenbaum's choice of words to describe Ershov, "his friend and host in Novosibirsk," to the audience of *Datamation*, where he published a short account of his 1964 trip to the Soviet Union: "My hat is off for this splendid and generous person." He added, "In the context of Yershov [sic], it is hard to understand what the Cold War is all about. The Soviet Union should send him abroad, under the Cultural Exchange Program, as their answer to Satchmo."[76] In his "From Minsk to Pinsk without Intourist," published in the "must-read" professional magazine of the day, the American expert combined a biting satire on the Soviet bureaucracy overseeing foreigners' movements and a skeptical vision of the Soviet computer industry with a praise of developments in programming and computer science.[77] His playful comparison of Ershov to Louis Armstrong, also pointed to an acute awareness of the public dimension of international scientific encounter and the cultural function of his own narrative. Feigenbaum had a message. His text cautioned the American computing community against a self-proclaimed superiority: "Undoubtedly the Soviet scientists make better use of their limited computer resources than we do of our surfeit."[78] The border-crossing computer scientist was a check on American self-importance.

The Big Science was public science, and circulating the press coverage of their visits to the guests was a common practice for Ershov and his American counterparts. In his June 21 letter to Ershov, Armer attached not only the print version of the article that appeared in *The Los Angeles Times* on June 8 but also the draft version before editorial changes. Although titled, "Vital Factor in Race: US Leading Russia in Space Computers," the content in fact included an overview of Ershov's visit: the IFIP meeting, a short description of Akademgorodok and his work on

76. E. A. Feigenbaum. 1965. From Minsk to Pinsk without Intourist. *Datamation*, 11(5): 34–36, on 35.

77. See R. V. Head. 2004. *Datamation*'s glory days. *IEEE Ann. Hist. Comput.*, 26(2): 16–21.

78. Feigenbaum, From Minsk to Pinsk without Intourist, 36.

computer languages, as well as characteristics of the new Soviet computer, BESM-6. The editorial changes took off the personal touch introduced by the journalist, describing how Ershov's spontaneous humor and wit captivated his audience.[79] The key argument of the article was left untouched: The current lead in computer technology should be no cause for complacency because "the USSR has first-rate mathematicians and engineers who do very well with what they have."[80] Armer's access to the draft was most probably a result of a personal contact with the journalist. The possibility of coordination regarding the tone and the message was also not unlikely given the affinity between the punch line of *The Los Angeles Times* article and Feigenbaum's message expressed in his talks, reports, and the *Datamation* article.

In this light, Robert Henkel's publication in *Electronic News,* discussed in the opening of this chapter, was not only a threat to Ershov but also compromised his hosts' vision for Soviet-American interaction. In other words, Henkel's article jeopardized Ershov's and the Americans' efforts on behalf of their respective national and international loyalties. Unable to repair what was done, Armer tried to mitigate the negative effects. On June 11, Armer sent an angry letter to the editor-in-chief of the *Electronic News* in order to protest the "inexcusable" behavior of their reporter toward a "distinguished" Soviet visitor.[81] The journalist chose to drop the descriptions of what Ershov had accomplished with the computers he had—and to focus instead on the characteristics of Soviet hardware and its many deficiencies described by the Soviet expert. But if the tone of the article and its usage of direct quotation were problematic enough, some of its information was not supposed to appear in the press in the first place. Ershov agreed to describe the non-secret Soviet series of Ural computers on the express condition to keep it "off the record," because he possessed no official permission for foreign publications about it. Henkel violated this agreement. The journalist, remembered as a hard-driving professional, preferred to reap the immediate benefits of new information for his newspaper, which belonged to Fairchild Publications and is known for its series, titled "Silicon Valley in the USA," that launched the term into public circulation.[82] At the heart of the exploding electronics industry, Henkel and his editors

79. EA: I. S. Bengelsdorf. US leading Russia in space computers. draft, f. 32/ l. 80–83.

80. Ibid.

81. FP: P. Armer to A. D. Cook. June 11, 1965. b. 26, f. 2, id: druid:xj638fz2416.

82. See the obituary Rob Lineback. February 19, 2014. Robert Henkel, early Silicon Valley journalist dies at 84. Available at www.eetimes.com/author.asp?doc_id=1321075. On the term, see D. C. Hoefler. 1971. Silicon Valley in the USA. *Electronic News*, January 11, 1971.

were not particularly sensitive to the question of international community building, the efforts it took, or the longer-term consequences of not keeping promises.

Armer had to assume the ACM's authority and responsibility as the organizer of the meeting. He threatened the editors of *Electronic News* to exclude the representatives of Fairchild Publications from ACM functions and in fact obtained Forsythe's agreement to bring the matter up at the ACM for a possible national action against the publisher.[83] Equally, Armer felt concerned about the possible consequences for Ershov and the germination of distrust. He decided to write a letter explaining the situation to Nikolai Kazarinoff, an American mathematician of Russian decent, and Ershov's friend since a previous long stay in Moscow. "I am very unhappy about the incident," wrote Armer, "and earnestly hope that it will not cause Andrei any embarrassment." Kazarinoff was in Moscow visiting the Steklov Institute of Mathematics and receiving his post via diplomatic channels, so he could transmit the letter and Armer's excuses to Ershov in person. Armer concluded his letter with a request for a first-hand report on Ershov's reaction without self-censure imposed by the Soviet surveillance system: "I would appreciate hearing from you Andrei's reaction to the unfortunate incident with the *Electronic News* reporter."[84] The letter showed a genuine concern on the part of the American expert and his reliance on personal trust and networks.

Armer was right to worry. Certainly, Ershov was not the only person to receive the press coverage of his trip. The Foreign Department of the Academy of Science was the agency responsible for traveling scientists and for what happened during their time abroad. Ershov had to pay an unpleasant "clean up" [chistka] visit to the party Central Committee at Akademgorodok. In addition, Ershov had to answer a personal letter from the president of the Soviet Academy of Sciences, M. V. Keldysh, known as the "chief theoretician" of the Soviet space program, requesting explanations. Keldysh's letter being lost, we have to rely on Ershov's response dated July 28, 1965.[85]

About one half of the letter was an almost line-by-line comparative analysis of the talk and its "voluntary" misinterpretation by the *Electronic News* reporter. Another was an argument for furthering international cooperation and exchange. The double translation between Russian to English and English to Russian became

83. FP: E. Feigenbaum to P. Armer. June 18, 1965. b. 26, f. 2, id: druid:vy010mh5767.

84. EA: P. Armer to N. Kazarinoff. June 24, 1965. f. 133/ l. 86–87. See R. Hersch. 1992. In memoriam: Nicholas D. Kazarinoff. *SIAM Review*, 34(1): 127.

85. EA: A. P. Ershov to M. V. Keldysh. July 28, 1965. f. 99/ l. 98–116.

a tool for transforming what was actually said by Ershov and what was heard by the audience into politically correct content. The main point of his talk, as presented to Keldysh, was to provoke the interest of the American specialists and to stimulate the exchange of materials on the cutting-edge of American research. Ershov declared to have gathered 165 titles but omitted that it was not during the ACM meeting that he gathered these materials. Next, Ershov turned the tables, and proceeded from self-defense to a counteroffensive advancing his vision of Soviet role in international cooperation in computing.

Ershov depicted an American computing community divided by their relationships with the Soviets. The American State Department and the mammoth companies, autonomous in their research, were behind the social order that represented Soviet computing as backward—an "uninteresting partner." This representation would be part of propaganda games against Soviet interests. Thus, the *Electronic News* article was just one instantiation of a larger attack. The friendly half of the American community consisted of the university academics and new small software companies, eager to cooperate and share their findings. So, what should be done: let the Americans believe in Soviet backwardness in order to misinform them? Or would it be better to counterattack by publicizing Soviet results? Ershov's arguments followed the logic of geopolitical competition. In Ershov's opinion, the first solution was "a luxury we cannot afford." To counterattack, a long term plan should be elaborated and realized through active participation in all computer-related international events, by using "*any* [sic] possibility to send our specialists to the US."[86]

Ershov's carefully crafted text brings together the contradictory co-dependency between the disciplinary and national loyalties. He obviously understood the operation of the Soviet science system. His letter was empowered by the high standing of his patrons—Dorodnitsyn, who was the official head of the Soviet delegation to the IFIP Congress, and Lavrentiev, the vice-president of the Academy of Sciences, who shared old personal bonds with Keldysh as members of Moscow mathematical school in the 1920s. The patronage networks do not explain the nature of the vision advocated by Ershov, however. His vision of Soviet computing was a product of personal reflection about national interests.

Ershov's official trip report is another expression of such a reflection, expressed through the technical descriptions of the American computing developments.

86. EA: A. P. Ershov to M. V. Keldysh. July 28, 1965. f. 99/ l. 115.

While the rules called for a document of a dozen pages, Ershov spared no efforts in sorting and summarizing all the information collected during the California part of the trip, resulting in a 200-page hand-written draft.[87] Ershov's "Report of the Trip to the US" includes a section on hardware, particularly the IBM 360 series and time-sharing. It also contains a section on software, covering the emergence of small companies, and, finally, a section on computer science education based on the materials from Stanford. The message was straightforward: making computing a state priority and fostering a closer integration with the American research agendas through international contacts and cooperation. Although not completely realistic, Ershov's ideas had important consequences.

While Ershov would never be able to accept Stanford's or any other Western invitation for a long-term visit, he would continue to travel abroad with increasing frequency throughout the rest of his career. Rising in the Soviet academic hierarchies, Ershov would become more and more involved in the state-level discussions of Soviet computing policy and institutionalization of the discipline of informatics, the position which would eventually make him one of the main advocates of the socialist version of the Information Society.

9.4 Conclusions: Divided Worlds, a Shared Community

From 1955–1965 a series of geopolitical crises brought the world to the brink of nuclear annihilation and the beginning of a new technological race that propelled humans into space. Computers were the new mathematical machines involved in the H-bomb calculations and the real-time antimissile defense systems, as well as orbit calculations for the space program. As argued by Paul Edwards in *The Closed World*, the computer became embedded in both political discourse and complex systems supporting the Cold War military frontiers.[88] However, by the middle of the 1960s, the computer also acquired a new dimension: it became a legitimate object of scientific inquiry. The new discipline of computer science was not contained by national borders or ideological differences.

The geopolitical divide and nuclear weapons systems prompted analogous developments on the both sides of the Iron Curtain. At the heart of the competition between the capitalist and socialist modernists were the new experts in search for a professional identity and the public recognition of their field, struggling with

87. EA: A. P. Ershov to A. A. Dorodnitsyn. September 25, 1965. f. 99/ l. 96.

88. P. N. Edwards. 1996. *The Closed World: Computers and the Politics of Discourse in Cold War America*. MIT Press, Cambridge, MA.

similar technical and social problems. Appropriating the discourse of geopolitical competition and working in a close cooperation with the military-industrial complex, they claimed an apolitical scientific authority. Such ambitions were predicated on an international coordination, the main function fulfilled by national and international professional societies, such as IFIP and ACM. These highly visible spaces of international encounters were maintained by two less tangible networks. Common interests, correspondence, and trust connected the experts on a personal level. Another crucial factor was the patronage of the state agencies interested in information collection.

If the Cold War surveillance regime was predicated on "knowing one's enemy," the state agencies received more than they asked for. The experts turned their access to information into leverage. Collecting information became an occasion to strengthen personal contacts and coordinate disciplinary visions. Transmitting information became an opportunity to argue for experts' vision for national policy and international discipline-building. Although the disciplinary history of the American computer science and Soviet informatics diverge due to the radically different national academic structure and the development of the computer industry, they were also entangled in a web of complex relationships. The two disciplines shared overlapping communities, visions of transformation for their national societies, and "universalisms" of mathematics and cybernetics.

When in New York, participating in the IFIP conference, Ershov applied for the membership in ACM. The paperwork went through, and he received the confirmation in July of 1965.[89] Ershov's ACM card—the last one was issued in 1987, a little more than a year before his death—was but a small token of his belonging to an international community of like-minded individuals. Friends and rivals, the Soviet and American experts kept meeting and learned how to preserve their community in time of crises. Belonging to the divided worlds, they did not spare any efforts to get closer to a shared future where computing technology would help humans to meet all material needs, solve ideological conflicts, and further knowledge about human nature.

89. EA: G. Forsythe to A. Ershov. July 9, 1965. f. 133/ l. 81.

10

Concern for the ‘Disadvantaged’: ACM’s Role in Training and Education for Communities of Color (1958–1975)

R. Arvid Nelsen

10.1 Introduction

The 1960s saw an increase in concerns expressed by computer professionals over social issues. In some cases the potential was for computers and/or computer professionals to provide positive solutions to social problems. In others, interest was in the impact of the computer industry itself and its role in creating or contributing to negative societal conditions. In the latter part of the decade, concerns focused on the plight of communities that computer professionals identified as “disadvantaged.” While the term “disadvantaged” itself meant different things, the communities were predominantly African American or Spanish-speaking residents of inner-city neighborhoods. Individual computer professionals who desired to be—and who saw themselves as—“involved” saw the potential of computing to alleviate joblessness and poverty in these communities. Thus, they developed programs aimed at providing basic education and training in computer operations and programming. While some programs originated within corporate environments or social welfare organizations, others were volunteer efforts by individuals in the computer profession, often working within their professional associations.

Members of the ACM acting within local chapters were responsible for some of these efforts, and the groundswell of interest ultimately led to institutionalization of the Committee on Computing and the Disadvantaged within ACM and the development of broader reaching initiatives. Volunteer members and local chapters could only accomplish so much. They required support—especially monetary support—minimally to ensure sustained progress but ideally to expand these training efforts. Institutionalization in this instance did not, however, result in either broader impact or sustainability. By 1972 while some chapters continued their involvement in such educational and job-placement programs, central ACM initiatives struggled. References to local activities in ACM publications eventually declined as well.

Factors leading to the general rise and decline of programs for the disadvantaged are manifold and complex. They include government initiatives, business interests, the changing economy, and the competition for attention by other emergent concerns. Additionally, programmatic efforts within the ACM were affected by a number of concurrent, related, and perhaps overlapping issues with which the organization and its members were involved. These include broad examinations of the social impact of computing, the problems of urban environments, the explosive growth of the computer industry, the rising demand for computer workers and the shortage of trained individuals, the cost of computing—including rising salaries for programmers, the nature and quality of private electronic data processing (EDP) schools, and debates on the status of computer workers as "professionals." ACM interests in such issues sometimes shared a similar life-cycle, rising to institutionalization in committees and special interest groups and eventually losing support and being de-chartered. Thus, an examination of ACM's activities addressing the disadvantaged must be understood within the context of related issues and efforts. In this book Janet Toland's chapter 6 on the ACM Special Interest Group on Computers and Society (SIGCAS) examines the association's various stands on "deeply political and social issues," focusing on significant developments from 1972 into the 1980s, including the Equal Rights Amendment (1972) and the plight of Soviet scientists as exemplified in the Turchin Affair (1975). Her investigations into the 1969 Question of Importance, concerns about Vietnam, and the formation of SIGCAS complement my own discussion of these topics in this chapter.

The sources for this chapter come from the documents of the ACM itself, primarily the flagship *Communications of the ACM* (*CACM*). *CACM* proved to be a rich source of information appearing in short pieces in the "News and Notices" and "Official ACM" segments. I focused on the period surrounding the active years for the training programs. Those years are 1968–1972, with the end of the ACM Committee on Computing and the Disadvantaged in 1974. In order to trace antecedents to these programs, I started my search in 1960. Due to the rather precipitous decline in references as early as 1972, I placed the terminus of my study at 1975. My study employed both in-depth review of content and text mining for the presence of keywords.

Publications of associated organizations provided information about the existence of programs not mentioned in *CACM* as well as details about them, through the publication of articles delivered and panel discussions held at conferences. These publications include: *Newsletter: ACM SIGCPR Computer Personnel*; *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*; *AFIPS '69 (Spring) Proceedings of the May 14–16, 1969, Spring Joint Computer Conference*; *AFIPS '69 (Fall) Proceedings of the November 18–20, 1969, Fall Joint Computer Conference*; *ACM '70: Proceedings of the 1970 25th Annual Conference on Computers and Crisis: How Computers Are Shaping Our Future*; and *SIGCPR '70 Proceedings of the Eighth Annual SIGCPR Conference*.[1]

Archival sources were also found in the Association for Computing Machinery Records (CBI 205) and the Edmund C. Berkeley Papers (CBI 50), both located at the Charles Babbage Institute at the University of Minnesota. Unfortunately, many details of the activities of ACM chapters and special interest groups (SIGs) and special interest committees (SICs) were not discoverable since the records of such ACM subgroups are not part of the corporate archive. Documents that discuss activities and controversies at the level of ACM Council typically summarized information, sometimes stating as little as the fact that a discussion on a group or topic occurred but offering nothing further. Thus, the "News and Notices" section of *CACM* was often the best source of documentation on these groups, their activities, and their interests. The rise and fall of actions by individual members and local groups appears related to the rise and fall of their discussion within these pages.[2] Attempts to supplement print and manuscript material with interviews have been partially successful. Information about one specific program was gathered via telephone

1. Other publications that have provided supplemental information but that have yet to be mined further include *Datamation*, *Computerworld*, and *Computers and Automation*.

2. It would be helpful to understand the mechanism by which editorial staff of the journal solicited, reviewed, and accepted submissions by chapters, as well as the relationship of the ACM Council or administrative staff to the editorial board of *CACM*, as any organizational control of the publication could affect the discussion or silence on any particular issue. For example, it is noteworthy that the creation of the ACM Committee on Computing and the Disadvantaged was widely announced in the journal, but its dissolution was never mentioned, while the decommissioning or de-chartering of other committees and SICs/SIGs was announced.

interviews with program leaders. Other leaders are deceased and family members were able to provide only fragmentary information.

Notably absent from the sources I have examined are the perspectives of the communities which were themselves the source of concern and the inspiration for action. Individuals who went through courses and were—or were not—placed successfully in jobs are not represented in the existing records and so a valuable perspective is thus far missing. The picture we get, therefore, is mostly from the perceptions of white professionals. We do have some documentation of programs from African American computer professionals who were leaders in this initiative, and while this is an important start it will be necessary to identify individuals who studied in and perhaps got their start through such programs in order to develop a more complete picture. My own ongoing research further traces the development and evolution of such out-of-school educational programs. It is an evolution that demonstrates the increasingly central role of persons and communities of color in the conception, development, and delivery of educational services and the story as a whole demands that their voices be heard.

10.2 The Discussion of Social Implications and Issues within the ACM

Questions pertaining to the role of computers in social problems, as well as the role of the organization in addressing such questions, arose among ACM members at various times. Opinions varied widely. Edmund C. Berkeley was one of the first advocates for the open consideration of issues of social responsibility by the Association. On June 2, 1958 he requested that the ACM Council appoint a committee to investigate, among other things, the question: "Do computer scientists have a special responsibility for helping to advance socially desirable applications of computers and helping to prevent socially undesirable applications of computers . . . ?"[3] Berkeley included reprints of discussions on the issue from his journal *Computers and Automation*, regarding whether the journal should engage such questions. The responses ranged from, "Computer scientists should realize their social responsibility and dependence on others," to "Nontechnical forums and periodicals exist for those who wish to make asses of themselves. Please stick to the subjects which are pertinent to computers, their applications, and allied fields."[4] The Council au-

3. Letter. To the Council of the Association for Computing Machinery. June 2, 1958. Edmund C. Berkeley Papers, Charles Babbage Institute, CBI 50, Box 9, Folder 22.

4. 49 computer people on SR of computer scientists. Reprinted from *Computers and Automation* September, 1958, Berkeley Papers, CBI 50, Box 9, Folder 22.

thorized the appointment of the Committee on June 11, 1958 when Berkeley made his case in person. Few mentions of the Committee appear in *CACM* over the years, and records documenting its existence have not been discovered in the records of the Association itself.[5] The Berkeley papers, however, contain many folders of correspondence and notes pertaining to these activities.

In the early 1960s, if mention was made of social issues in the pages of *CACM*, the prevailing concern was that of unemployment and technological displacement due to automation. Discussion of a particular issue did not suggest any one particular viewpoint. Contrary perspectives are evident on the issue of automation, for example. The summary of a presentation made at the January 21, 1960 meeting of the Washington, DC chapter by Ted F. Silvey from the AFL-CIO Research Division, "Labor Views Automation," attempts a balanced view. Silvey stated that, ". . . trade unions welcome automation, since the new technology enables mankind to overcome its age-old twin enemies: drudgery and poverty. . . . Automation is neither moral nor immoral, but amoral. It is people who are moral or immoral; it is what people do with automation that makes it good or evil."[6]

CACM also reported on interest by government in the issue of automation. "White House Conference on Automation" was a March 1962 *CACM* headline.

> Senator J. Caleb Boggs (R-Del.) introduced a bill (S. 2772) Jan 31 calling for a White House conference on the impact of automation . . . The bill states that the primary responsibility for meeting the challenge and problems of automation is that of commerce, industry, labor, and local communications.[7]

On January 11, 1962, President Kennedy's State of the Union address mentioned automation only once but clearly expressed concern about negative impacts: "Too many unemployed are still looking for the blessings of prosperity. As those who leave our schools and farms demand new jobs, automation takes old jobs away."[8]

5. The single sentence, "A Committee on Social Responsibilities rendered its first report to the Council in March 1959," appears in F. L. Alt. 1962. Fifteen years of ACM. *Commun. ACM*, 5(6): 300–307. The committee is also mentioned in the biography of Saul Gorn, who lists himself as a past chairman of it when he stood for Member-at-Large in 1962 (*Commun. ACM*, 5(3): 131) and again in 1966. In April 1967 Paul Armer is referred to as chair of the ACM Committee on Social Responsibilities (*Commun. ACM*, 10(4): 248). The committee does not appear in official committee directories in the 1960s.

6. News and notices: ACM chapter news. *Commun. ACM*, 1960, 3(3): 176–177.

7. White House conference on automation. *Commun. ACM*, 1962, 5(3): 180.

8. John F. Kennedy. Annual Message to the Congress on the State of the Union, January 11, 1962. Found on the website of *The American Presidency Project*. Available at http://www.presidency.ucsb.edu/ws/?pid=9082 (accessed January 2016).

In response, the President's Advisory Committee on Labor Management Policy expressed support for automation "within hours of the President's State of the Union Message to Congress." The Advisory Committee stressed to the President: "that automation and technological progress are essential to the Nation's well-being; that this progress can be achieved without the sacrifice of human values . . . We reject the too common assumption that continuing unemployment is an inherent cost of automation."[9]

As the decade proceeded the content of ACM conference presentations suggests that potentially negative impacts of computer were taken more seriously. In December 1964, the Washington, DC chapter of ACM announced the formation of its own Special Interest Group on the Social Implications of Information Processing, stating that one goal was to "objectively discuss the social effects of automation." Chaired by Peter Warburton of Control Data Corporation, the group proposed a number of questions:

> Does automation destroy jobs? Or does it create jobs? Does it upgrade society? Or does it downgrade the individual? Does it preclude creativity? Or does it release human energy for more creative activity?[10]

The 1964 ACM annual conference bore the title "Computers '64: Problem-Solving in a Changing World" and its publicity promised "sessions on automation and the social implications of information processing technology." The description went so far as to say, "It has been pointed out that the year 1964 may be significant by being the midpoint between the infancy of electronic digital computers . . . and the year 1984, designated by many critics as the year when the world will be completely infused with the potential evils of our advancing technology."[11] The call for the 1964 AFIPS Fall Joint Computer Conference advertised a symposium on "Plans, Procedures, and Constructive Suggestions On What To Do To Minimize the Dislocations To Be Expected in the Transition from the Society of Today to the Computer-Automated Society of the Future."[12]

Invited speakers also engaged the subject of social issues and the role of the computer professional. The 1967 ACM conference included a keynote address by

---

9. The President's Advisory Committee on Labor-Management Policy. *Commun. ACM*, 1962, 5(1): 73.

10. *Commun. ACM*, 1964, 7(12): 741.

11. *Commun. ACM*, 1964, 7(1): 49.

12. *Commun. ACM*, 1964, 7(9): 562.

Emmanuel Mesthene, director of Harvard University's program on technology and society, who stated:

> . . . it may no longer be the duty of the scientist to pursue truth regardless of where it leads, but that science must take into account the tailoring of its procedures if disastrous results are to be avoided. He stated that ACM's growing concern for the social consequences of the professional work of its members is indicative of a transition from tradeunionism [sic] to professionalism.[13]

On May 16, 1968, the Washington, DC chapter held a technical symposium titled "The Technology Gap," at which Herbert R.J. Grosch (then ACM Council member and director of the National Bureau of Standards Center for Computer Sciences and Technology and later ACM president 1976–1978) is reported to have stated: "the real measure of the computer profession is what it does to close the gap between solving "clean" scientific problems and solving "messy" social, political, and economic problems that beset our country today. . . . we can no longer say about social problems 'It is not our problem'."[14]

But even as individual members, speakers, and chapters engaged with social issues, when the matter of ACM's engagement as a professional association was raised, its leadership wrestled with both the organization's ability to do so and the pertinence of such matters to the ACM's mission. President Anthony Oettinger's March 1967 letter to ACM membership discussed the importance of professional concern over private EDP training programs. Yet, Oettinger's overall message was that boundaries needed to be placed. The letter began boldly:

> Social responsibility is a topic that ACM members, like much of the community, are fond of discussing these days. Whatever our thoughts and actions might be as private citizens, some questions of social responsibility are irrelevant to the ACM and its members (as ACM members) because we have no special collective competence to deal with them while there are others whose solution critically depends on our interest and active cooperation. It is the latter to which this letter is addressed. . . . Action or inaction by ACM as a body or by individual members in their professional capacity is too important to be left to the impulses of bleeding hearts or cynics. Reality has a nasty way of being many-sided.[15]

13. Mesthene on technology and society. *Commun. ACM*, 1967, 10(10): 672.

14. Social problems focus of DC chapter annual technical symposium. *Commun. ACM*, 1968, 11(7): 527–528; Computing pioneer Herbert Grosch, dead at 91. *ACM News* January 29, 2010. Available at http://cacm.acm.org/news/70203-computing-pioneer-herbert-grosch-dead-at-91/fulltext.

15. A. G. Oettinger. 1967. President's letter to the ACM membership. *Commun. ACM*, 10(3): 139–140.

Oettinger then defined three specific "major political and social issues" that he felt demanded the attention of the ACM Council: the relation between computers and communications, citing a recent inquiry by the Federal Communications Commission (FCC); professional ethics, citing a recent Federal criminal case against an individual accused of manipulating computer data to defraud a bank; and the previously discussed question of private EDP training facilities and the attempts of the ACM education committee to address problem schools and promote the accreditation of computer education. Thus, Oettinger did not claim the irrelevance of social concerns, but he did limit involvement to issues that he saw as central to ACM's core interests and competencies.

The argument of his letter hinged on a conception of the Association's mission and interests. A year later, in May 1968, Oettinger turned to the matter of the Association's *ability* to speak on social issues, practically and legally. Striking a balanced tone, Oettinger allowed that "it would be most irresponsible to ignore problems of public policy and to behave as technicians totally unconcerned with the consequences of their work." He emphasized the diversity of opinion among ACM's membership, however, and questioned the ability of the Association to maintain a unified position on controversial subjects:

> There are, however, many diverse points of view on what ACM, as an organization, should do about it. . . . Few of us agree with one another on technical issues, and perhaps still fewer would agree on assessments of the implications of our technological products. The ACM should therefore not presume to speak with a single voice on technical or social issues, since no officer can pretend to represent "the membership's views" . . . [16]

Regarding legality, he invoked Section 501(c)(3) of the Internal Revenue Service Tax Code, citing the restriction on political speech. For such organizations, "no substantial part of the activities of which is carrying on propaganda or otherwise attempting to influence legislation." Oettinger did suggest that there are things that members could do *as individuals*, offering as an example his own personal testimony before a Senate committee.[17]

16. A. G. Oettinger. 1968. President's letter to the ACM membership. *Commun. ACM*, 11(5): 293–294.

17. The ongoing relevance to ACM leadership of these specific arguments against the ACM's ability to comment on social issues (the diversity of members' opinions and the association's 501(c)(3) tax status) can be seen in Toland's chapter 6.

Later that year, a different message was delivered by ACM's next president, Bernard Galler. His October 1968 president's letter bore the title "Involvement." The letter was a call to action:

> Probably there is no profession where one gets more deeply involved in his work than ours. . . . But it is probably that same involvement that tends to keep us from looking out and realizing that we're still part of the larger world. While most of us are undoubtedly sympathetic to the problems of the poor, the hungry, and the jobless, I dare say most of us, like the general population don't do very much toward solving these problems. It is an interesting (and often discussed) question whether we as technical professionals have a stronger obligation toward society than others; on this point I could no doubt fill this letter with a philosophical discussion. The fact is, it is almost irrelevant whether we have more of an obligation than the general community. We, like others in the general community, have an obligation that we aren't beginning to fulfill.[18]

He recounted his own participation in the ACM Special Interest Committee on the Social Implications of Computing (SIC)^2 in tackling the question of how computer people might get more involved, as well as a collective desire for "averting crisis." Examples of suggested actions included "clean up part of a slum neighborhood in Detroit" and "the education of underprivileged dropouts." Both the call to action and the list of proposed concerns differed from the narrower focus proposed by President Oettinger, and yet it remained close to Oettinger's call for *personal* action. In regards to actions within the association, Galler called attention to the Committee on the Professional Activities of the Blind and the then-newly formed Committee on Computers and the Disadvantaged. He also mentioned the Computer Industry Martin Luther King Fund, although it is unclear whether this was officially connected with the ACM.

The question, however, of whether or not the ACM and affiliated associations should be more involved and vocal *as organizations* persisted. The February 1969 issue of *CACM* included an article stating that the ACM Council had received a petition "requesting that ACM should go on record as firmly opposed to the war in Vietnam." The Council decided by vote that the action requested fell outside of the purpose of the ACM and its constitution and bylaws. In an interesting move, however, the Council decided to put a vote to the general membership on "a Question of Importance to determine the preference of the membership regarding the extent of the purposes of ACM, namely, should ACM express positions on

18. B. A. Galler. 1968. President's letter to the ACM membership: Involvement. *Commun. ACM*, 11(10): 659.

political issues."[19] A letter to the membership from President Galler dated April 28, 1969 included two Questions of Importance, the first on a proposal to move the location of the 1970 conference, and the second which read, "Shall the Constitution of the ACM be revised to permit Association comment or action on deeply political and social questions?"[20] The letter included the history of the Council's receipt of the petition and the details of their vote, as well as pro and con statements for the revision to the constitution, and the text of the petition itself. The July 1969 issue of *CACM* reported that the measure was voted down, with 2,059 in favor and 7,938 opposed.

As printed in the ballot, this petition does not identify the person, persons, or group that drafted it. A separate resolution, however, published in the April 1969 issue of *Computers and Automation* and the March 1969 issue of *Interrupt*, the newsletter of the group Computer Professionals for Peace (CPP), explicitly identified the drafters as members of the ACM Special Interest Committee on the Social Implications of Computing (SIC)^2: "As professionals in the computer field and members of (SIC)^2, we have a responsibility through our professional association, to oppose the use of our skills for destructive and anti-social ends."[21]

Further complicating the picture, this issue of *Interrupt* included an article titled "The A.C.M." in which CPP protested the recent announcement that President Galler had dissolved (SIC)^2 on the grounds of inactivity. The column claimed that the SIC had been formed "despite opposition" and that, "The Association for Computing Machinery has long maintained that as "professionals," we need not be concerned with the morality and social implications of our work." It further asserted that, "recently several members of (SIC)^2 in the New York area got together and wrote a paper which, we think, effectively demolishes the A.C.M.'s pretense of professional neutrality."[22] Additionally, the April editorial in *Computers and Automation* disputed the authority of the ACM president to single-handedly dissolve a SIC and asserted, "Accordingly, SICSIC still exists, because his actions are null and void."[23]

19. Should ACM consider positions on political and social issues? *Commun. ACM*, 1969, 12(2): 121.

20. Memorandum. April 28, 1969. Association for Computing Machinery Records, CBI 205, Box 3, Folder 18.

21. Resolution. *Computers and Automation*, 1969, 18(2): 13; Resolution. *Interrupt*, 1969, p. 11. The full text of the petition as published in *Interrupt* can be found in Toland's chapter.

22. The A.C.M. *Interrupt*, 1969, p. 1–2.

23. *Computers and Automation*, 1969, 18(2): 13; Resolution. *Interrupt*, 1969, p. 14.

Tensions may have increased even further. Computer Professionals for Peace announced the group's intention to attend the AFIPS Spring Joint Computer Conference in May 1969. The July 1969 issue of *Datamation* included a dramatic description of the conference:

> For perhaps the first time at a JCC, the computer industry made *loud* noises—some wailing, some obscene, but most important, some constructive—about its concern for the World Outside the Technology and Business of Electronic Data Processing. . . . There were the protests against the involvement of the industry and the universities in war and other military efforts. And there was the plan to re-establish a social implications committee of the Association for Computer Machinery, perhaps portending more real-world projects—yet undefined—by that association.[24]

The keynote address by General James M. Gavin "set the stage for the social implications emphasis of the conference" which included presentations on computer training programs for the disadvantaged. The article also described a session at which Computer Professionals for Peace was present, titled "URGENT—An Increased Dialog with Society" as the "most emotional" due to the CPP's desire to discuss the war. "Their shock tactics, in the form of repeated interruptions and some obscenities, did indeed unnerve the philosophical panel and stir the audience to its feet. But little constructive action resulted—except, of course, to bring the aims of the CPP to the attention of the audience, and to the doves this was enough success."[25]

The actions of the CPP appears also to have caused a reaction from ACM leadership. On June 5, 1969, Donn B. Parker, secretary of the ACM, wrote to the ACM Council listing five topics discussed at the ACM Executive Committee meeting held three days earlier. The second topic reads: "Control of Militant Activists at conferences. Parker asked for comments on proposed policy. Legal council is to be consulted on conference liability."[26]

Although their concern about Vietnam and the military application of computer technology is possibly their best known activity, CPP also promoted the cause of

24. The SJCC: Boston will rise again. *Datamation*, 1969, p. 59. Emphases in italic and capital letters are original to the article.

25. Ibid. pp. 59, 61.

26. Association for Computing Machinery Records, CBI 205, Box 6, Folder 29.

civil rights. The same issue of *Interrupt* that criticized the ACM's decision to dissolve (SIC)[2] included several relevant entries. An article on the Computer Personnel Development Association (CPDA) explained that its focus was on disadvantaged inner-city communities of color (discussed at greater length below). Pages 5–7 under the heading of "Racism in the Industry—a forum" include two articles laid out side-by-side: "Peace and Black Liberation" by Henry Warfield and "The Issues of War and Racism" by Joan Manes and Edward Elkind. The CPP's activism on this front was reported in *CACM*. The January 1971 article "Council Decides Against Support in Clark Squire Case" reported on the CPP's efforts at the 1970 ACM conference to obtain the support of the association in raising bail for a computer programmer who had been arrested along with 20 other members of the Black Panthers in New York City. The ACM did not grant this support. The article stated that, "the Council has issued a statement that ACM action was outside of its constitutional purposes," an argument already discussed above.[27]

These controversies over social issues serve as a reminder that the activities of the individuals and groups involved with educational training for disadvantaged communities did not arise in a vacuum. The history of the association shows that interest in the broader society and its intersections with the areas of research and industry central to the ACM and its members varied over time and that opinions were diverse. Furthermore, the specific period in which activity on behalf of the "disadvantaged" occurred was marked by significant controversies. In addition to other factors, it's possible that simple overload and fatigue began to plague ACM leadership.

10.3 Programs for the "Disadvantaged"—1968–1972

The late 1960s saw a rise in highly focused and job-oriented training programs aimed at placing Black and Spanish-speaking persons in entry-level positions as keypunch operators, computer operators, or programmers. Training opportunities were created specifically for residents of urban neighborhoods, often described as "inner-city" or "ghettos" in order to help members of communities identified as "disadvantaged" escape from joblessness and poverty. These programs were not created as part of traditional K-12 or post-secondary schools and were distinct from the offerings of private electronic data processing (EDP) schools.[28] Rather they were developed by local chapters of computer professional associations, social

27. Council decides against support in Clark Squire case. *Commun. ACM*, 1971, 14(1): 55.

28. Regarding K-12 education, see W. Aspray. 2016. *Participation in Computing: The National Science Foundation's Expansionary Programs*. Springer International, Basel. For discussion of private EDP

welfare organizations, and state or local government. Support occasionally came from the business community and a few programs were in-house developments by individual companies. Programs were predominantly volunteer efforts; even those developed within business environments were sometimes said not to be official and the individuals involved may have functioned as volunteers.

Although individual programs were started autonomously by separate organizations acting in isolation, they appear to have developed at roughly the same time. On Wednesday, November 19, 1969, at the AFIPS Fall Joint Computer Conference, David B. Mayer of the IBM Systems Development Division declared:

> Motivated computer professionals all over the United Stated have undertaken a most special and extraordinary task: they are involving themselves in every way possible in the training of disadvantaged and educationally-deficited men and women from the so-called ghetto and poverty areas of the country. They are exhibiting a special and wonderful tension which impels them to appear at that interface between their own computing community and those underprivileged who wish to enter it.[29]

Mayer provided a summary evaluation of 11 projects established between the summer of 1968 and September 1969.[30] I have identified seven others, bringing the total to 18 that appear to have operated between 1968 and 1972. Most were located on the East Coast or in California, with one in the Midwest. Cities offering these opportunities included: Boston; New York City; Schenectady, NY; Philadelphia; Washington DC; Baltimore; St. Louis; San Francisco; Menlo Park; Los Angeles; Sacramento; and La Jolla/San Diego.

Although descriptions of programs were varied, race was central to their purpose. Writers reporting on programs almost always referred to their target audience as "disadvantaged" or "underprivileged." On its own, the term "disadvantaged" need not have any particular connotation. A later and broader definition of disadvantaged communities would appear in the transcript of a panel discussion titled "Computing and the Disadvantaged" from the 1970 ACM annual conference. This panel included discussions of the physically handicapped with specific

---

schools, see N. L. Ensmenger. 2012. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*. MIT Press, Cambridge, MA.

29. D. B. Mayer. 1969. The involved generation: Computing people and the disadvantaged. *AFIPS '69 (Fall) Proceedings of the November 18–20, 1969, Fall Joint Computer Conference*, pp. 679–690. ACM, New York.

30. He observed or participated in 10, but his appendix lists 11.

reference to the blind, convicts, minorities, the poor, and "hardcore unemployables."[31] In respect to the programs examined here, however, while reports often emphasized the economic conditions of the communities that programs intended to serve they quickly turned their attention to race. In his November 1969 presentation, Mayer stated that the term "disadvantaged" originally described "youths from poverty backgrounds" but had been "broadened to include all those who are educationally-deficited . . . including those from both poor white and non-white communities."[32]

Nevertheless, program details focused specifically on the intersection of race and poverty. Mayer's analysis identified 11 characteristics of program participants, the third of which read "are black or brown." *CACM*'s description of a program at Johns Hopkins University read: "Five male students from a Baltimore 'inner city' high school enrolled and completed the course of instruction. The purpose of the program was to train otherwise unskilled black youths for beginning positions in the computer field."[33] For the program in Schenectady, "The only restriction on entering students was that they must be black."[34]

Yet if "disadvantaged" at this time came to be restricted to persons of color, discussions of communities of color were similarly restricted to those from economically disadvantaged backgrounds—signally failing to acknowledge individuals who had achieved higher levels of educational and professional accomplishment.[35] Although women and men of color appear infrequently in computing publications in this period, publications by communities of color often profiled individuals working in computing fields, such the "Speaking of People" column in *Ebony* magazine which featured six individuals from different professions each month.[36]

Similarly elided from presentations on these programs are persons of color who provided actual instruction. Walter A. DeLegall, an African American computer professional at Columbia University, wrote:

31. V. Henriques. 1970. Computing and the disadvantaged, a panel session. *ACM '70: Proceedings of the 1970 25th Annual Conference on Computers and Crisis: How Computers Are Shaping Our Future*, pp. 8–15. ACM, New York.

32. Mayer, The involved generation, 679.

33. Computer training program for Baltimore 'inner-city' youths. *Commun. ACM*, 1968, 11(12): 867.

34. A. J. Bernstein. 1971. A programmer training project. *Commun. ACM*, 14(7): 449–452.

35. See J. T. Barber. 2006. *The Black Digital Elite*. Praeger, Westport, CT.

36. R. A. Nelsen. 2017 (forthcoming). Race and computing: The problem of sources, the potential of prosopography, and the lesson of *Ebony* magazine. *IEEE Ann. Hist. Comput.* Article as preprint available at DOI: 10.1109/MAHC.2016.11.

> Invariably we come to the consideration of the importance of ethnic background in the selection of teachers for training programs aimed at minority group members. Is a Black teacher better equipped to teach Black students? All else being equal, I would say yes. The chances of a compatible student-teacher relationship are maximized if the teacher has some intimate association with the problems and life styles of his students.[37]

DeLegall's assessment seems to be reflected in the actual practice of several programs. One operated at the Lowell School reported that most teaching assistants assigned to the "hardcore deprived" were black. The Computer Jobs Through Training program at San Diego employed as its chief instructor, Robert W. Sadler, an African American scientific programmer. Other instructors were computer science students at UCSD, many of whom were Chicano.[38] Beyond brief mention of such participation, however, articles discussing Mayer's "motivated professionals" apparently assumed they were white. Mayer himself wrote, "The first psychological jolt for motivated whites is to discover they will not, most of them, be acting at the actual 'interface' between the disadvantaged and the advantaged community."[39]

Thus, although educated and professional persons of color played central roles in training, discussion of communities of colors focused mostly on characteristics of economically and educationally deprived communities and occasionally explicitly conflated economic disadvantage with racial identity.

This construction of perceptions of communities is significant because it appears to have played a part in the choices made regarding program content and structure. Program reports frequently expressed the need to set achievable goals based on the capabilities of target communities. Without the counterbalancing influence of acknowledged computer professionals of color, educational deficiencies could become equated with being Black or Latino, rather than with the deprivations suffered by communities defined by geography and wealth. Even among the "motivated whites" called to serve disadvantaged communities, sensitivities varied in the ability to distinguish between aptitude and academic achievement, as can be seen in the approaches taken by two different programs in regards to prospects in programming. The Computer Personnel Development Association developed a

---

37. W. A. DeLegall. 1969. Teaching techniques and quality education/training for the 'disadvantaged'. *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*, pp. 91–94. ACM, New York.

38. Author's telephone interviews with M. Granger Morgan (February 21, 2015) and Robert W. Sadler (March 25, 2015).

39. Mayer, The involved generation, 688.

computer operator training program. When students inquired about their potential to move into programming, "They were . . . warned that to move from a computer operator to programmer they would need at least a full high school diploma, if not some college-level training."[40] Conversely, providers of a FORTRAN training program at General Electric in Schenectady stated, "No testing was done, a high school diploma was not required, and a police record was no bar. . . . Students were chosen on the basis of aptitude and intelligence. . . ."[41] The content of the eighteen courses that I identified indicates that opportunities for students to pursue computer operator or programmer positions through this type of training were roughly equal, at least nationally if not locally. Nine programs offered some course in computer programming. The specific course content was often not mentioned in brief reports, but when it was included the courses offered were in either COBOL or FORTRAN. Eleven programs offered courses in computer operations. Specific hands-on experience included IBM 1401, IBM 360/30, IBM 360/40, IBM 360 DOS, and IBM 7030 (Stretch). Four programs offered courses in keypunch operation.

Program providers similarly varied in their approach to the impediments faced by students coming from economically disadvantaged backgrounds. In respect to impediments, members of the Urban Education Committee of Philadelphia wrote:

> The Training and Curriculum Committee also recognized that in many cases a student would have social problems which might interfere with his ability to function to his full capacity. . . . It must be impressed upon the student that any social problems that he might be experiencing cannot be an excuse for doing poorly. . . . The social aspects of a student's life, regardless of how unpleasant, must be handled in an adult way and not be used as an excuse for under-accomplishment.[42]

Conversely, the Computer Jobs Through Training program in San Diego attempted to mitigate obstacles to learning:

> Economically the simplest approach is to take the students to a central training facility. . . . in San Diego, while the Black community is somewhat localized, the Chicano or Mexican American community is spread all across the city in a

40. J. P. Gilbert and D. B. Mayer. 1969. Experiences in self-selection of disadvantaged people into a computer operator training program. *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*, pp. 79–90. ACM, New York.

41. Bernstein, A programmer training project, 450.

42. H. Griffin and P. Gravelle. 1969. Report on the philosophy and mechanics of the Urban Education Committee of Philadelphia. *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*, pp. 95–98. ACM, New York.

> collection of widely spaced communities. In the early portions of the course, motivation is the single most important consideration—and one good way not to motivate people is to make them sit on a bus for an hour or more every day riding to and from a class. . . . The solution we chose was a mobile instructional facility housed in a forty foot trailer truck . . . [43]

Computer Jobs Through Training also demonstrated a greater willingness to meet the language needs of its Spanish-speaking community:

> All of the early material in the course is available in English on one side of the page, and Spanish on the other. Clearly no programmer can be placed in a job in this country if he is not fluent in English. Fluency in English is a prerequisite for entrance to the course. But being fluent in English and being comfortable in English are two different things. During the 1969 Youth Corps course we found that several of our Chicano students became much more interested and did much better work when problems were available bilingually and when instructors showed a willingness to use Spanish.[44]

The CPDA program in New York similarly reported using bilingual instructors.[45]

Authors placed the activities of the programs on which they reported within the broader social context of the time, explicitly addressing the increase in civil unrest in many cities. The specific motives of the "motivated computer professionals" was sometimes questioned. Were they genuinely altruistic or were they self-interested? Speaking on a panel titled "Computers and the Underprivileged" at the Spring Joint Computer Conference in May 1969, Milton Bauman cited "these days of urban crisis" and the goal of getting "the disadvantaged to become involved in the mainstream of American life."[46] One month later DeLegall expressed cynicism about the motivations of these training efforts: "Often in a genuine effort to help, but occasionally for commercial reasons or simply to de-fuse and dissipate the burgeoning discontent of these communities, training programs for the 'disadvantaged' minorities have sprung up."[47]

43. M. G. Morgan, M. R. Mirabito, and N. J. Down. 1970. Computer jobs through training: A preliminary project report. *AFIPS '70 (Fall): Proceedings of the November 17-19, 1970, Fall Joint Computer Conference*, p. 346. ACM, New York.

44. Ibid., 349.

45. Gilbert and Mayer, Experiences in self-selection, 89.

46. M. Bauman. 1969. Computers and the underprivileged. *AFIPS '69 (Spring) Proceedings of the May 14–16, 1969, Spring Joint Computer Conference*, pp. 35–36. ACM, New York.

47. DeLegall, Teaching techniques and quality education/training for the 'disadvantaged', 91.

In 1972 Cecil Page Brown reviewed many of these programs and referred specifically to "the months following the periods from Summer, 1964–Summer, 1967, in which the country experienced its gravest civil disorders in modern times" and asserted, "In short, jobs and employment opportunities were being offered to the disadvantaged as a step toward curing social ills."[48] In a July 1968 article "Employing the Unemployable" *Fortune* magazine quoted Charles Y. Lazarus, head of the American Retail Federation:

> How was it possible, he asked, that "a whole generation of corporate executives could drive their Cadillacs through the slums of America and not know that someday these ghetto-poor would rise to threaten both the corporate balance sheet and the whole fabric of American life?"[49]

The federal government also reacted to increasing incidents of civil unrest. On July 28, 1967, President Lyndon B. Johnson established the National Advisory Commission on Civil Disorders. The Commission's report of February 1968 described major sites of unrest, including Newark, Detroit, and New Brunswick, and citing the conditions of inner-city communities, poverty, and unemployment among the factors contributing to unrest, as well as the nation's long history of slavery, racial discrimination, and segregation. Government responses to the situation included the formation of initiatives designed to provide incentives to the business community to create training and employment opportunities, including the Concentrated Employment Program, Job Opportunities in the Business Sector, and the National Alliance of Businessmen.

If civil unrest and governmental incentives provided impetus for businesses to support efforts of some kind, the "commercial reasons" hinted at by DeLegall may suggest how such efforts were shaped. While he did not go into additional detail, we might look to the work of two historians to understand what such reasons might be and their effect on shaping programs. In *Race on the Line*, a perceptive analysis of race and gender in the Bell System, Venus Green describes the company's participation in the National Alliance of Businessmen and then Vice President Johnson's 1962 "Plan for Progress." Green concludes that such efforts were primarily public-relations ploys that simultaneously developed an inexpen-

48. C. P. Brown. 1972. An investigation into the recruitment and training of the disadvantaged for staffing the MIS function. *Computer Personnel: A Quarterly Publication of the Special Interest Group on Computer Personnel Research*, 3(2): 3–8.

49. C. Burck, editor. 1968. Employing the unemployable. *Fortune*, 78(1): 29–30, 34.

sive, segregated, and controllable work force. In *The Computer Boys Take Over*, Nathan Ensmenger describes the growing need for computer programmers in the late 1960s, as well as the increase in salaries and degree of power they commanded, a situation that some businessmen found undesirable. A piece in the July 1965 issue *CACM* bore the headline: “Programming Costs Too Much.” The article was a report by Jim Tupac of the RAND Corporation on a panel held at the IFIP Congress ’65 titled “Economics of Programming.” Among the contributing factors were “rising salaries” and “growing demand for personnel.” Possible solutions suggested included, “better selection techniques and improved training and educational programs . . . to meet the growing demands for personnel at all levels.” By 1968, the ACM acknowledged the role such interests played in the formation of the private EDP schools. On January 3 of that year, ACM President Oettinger appointed Carl Hammer as chairman of a newly chartered ACM Accreditation Committee. An October 1968 article stated that, “the Committee is fully aware of the fact that hundreds of institutions have been opened in response to heavy demand for minimally trained personnel.”[50] The creation of structured entry-level positions, limited in scope and requiring minimal training, had the potential to provide a source of inexpensive workers with much more limited power.

Thus, it was within a context of dramatic civil unrest and both government and corporate incentives that these largely volunteer training programs were created across the country. While they may have been conceived and offered by volunteers, they often relied on government agencies and businesses for grants and donations of money and materials. In order to successfully place graduates, they specifically required buy-in from the business community. As early as Mayer’s 1969 presentation, financial challenges were apparent. He wrote, “The single greatest lack is funds” and that volunteer projects “tend to peter out.” Reports of specific programs almost always appear at the beginning of programs, as they were kicking off or after a first class had been completed. The only published terminal report that I found is a 1972 article on Computer Jobs Through Training, a program developed in San Diego by graduate students at UCSD. The article stated:

> The single most overriding problem which has characterized this project almost from its beginning has been the inadequacy of funding support . . . Without

50. Recommended ACM position on accreditation involves support of existing agencies. *Commun. ACM*, 1968, 11(10): 714.

> long-term continuity in program support, it becomes impossible to perform many functions efficiently . . . if at all.[51]

In addition to funding problems, the UCSD program also experienced difficulties in job placement. The final report stated that the recession and growing unemployment was accompanied by an increase in expectations for entry-level positions. Motivated individuals were not enough to ensure sustainability.

10.4 ACM Involvement

During the brief period in which these efforts flourished, the ACM, its members, and *CACM* played significant roles. *CACM* reported the activities of individuals and groups around the country who were involved with this and other social issues, both within and outside of the Association itself. Local ACM chapters sponsored lectures on relevant issues and training programs offered by others. A few became directly involved, either establishing their own educational programs or contributing support to existing programs. ACM became involved at its national level in two ways, the first being involvement by the Special Interest Group on Computer Personnel Research (SIGCPR), followed closely by the establishment of the Committee on Computing and the Disadvantaged.

Few references to communities of color appear in *CACM* in the early 1960s. The earliest is in an article summarizing a panel discussion that occurred at the 1964 ACM national conference, "The Computer Revolution: Its Effect on U.S. Business and Labor." One panelist, Carl Linder, assistant to the president of the Glass Bottle Blowers Association, AFL-CIO, "pointed to the fact that the Negro, who comprises only 11 percent of our population, accounts for 22 percent of the unemployed."[52] The next explicit mention of communities of color occurs in 1967—perhaps not coincidentally—within discussion of the growing concern felt by computer profes-

51. M. G. Morgan, N. J. Down, and R. W. Sadler. 1972. Computer jobs through training: A final project report. *1972 Fall Joint Computer Conference, December 5–7, 1972. AFIPS Conference Proceedings* 41, Part II, pp. 1243–1249. American Federation of Information Processing Societies, Inc., Montvale, NJ.

52. M. Rubinoff. 1964. Impact of computers on society. *Commun. ACM*, 7(10): 637–638. Additional discussion of the disproportionate effect of automation on African Americans can be found in a speech delivered by Whitney M. Young, Jr., Executive Director of the National Urban League, to the Conference on Employment Problems of Automation and Advanced Technology in Geneva, Switzerland on July 20, 1964. The intermingled revolutions: The Negro and automation. *Vital Speeches of the Day*, 30(22): 692–694.

sionals over private EDP training programs. In the March 1967 "President's Letter to the ACM Membership" Anthony Oettinger addresses three prevailing concerns worthy of consideration by the association. One of these pertained to private "computer programmer schools." Oettinger quotes extensively from a letter by the chairman of the ACM Committee on Professional Activities of the Blind, Theodore Sterling, who characterized the schools as "little more than rackets" that "solicit heavily among poverty stricken sections of the population." A particular example provided by Sterling reads:

> This school (in Cincinnati) first came to my attention because a Negro with practically no education or literacy came to us to inquire about a job. He had paid a fee of $750 to the school under the impression that he was going to be trained as a programmer. He learned, as far as I was able to make out, to run a sorter and to do minimal wiring on an IBM 407. While his contract did not state that he would be trained to become a programmer, he certainly was under the impression that this was what he had paid for. At the conclusion of the course he received a rather gaudy certificate and was encouraged to enroll into the next and more advanced class (at a fee of $1,100) which he was shrewd enough to decline. His only reward for a $750 fee was the certificate (with prominent gold seal) and an unrealistic list of possible employers.[53]

Thus, we see that the professional community was increasingly aware of and concerned about the presence of such schools, in additional to general awareness of stresses felt in poor urban environments.

1968 marks a transformative year: training programs were established and the pages of *CACM* begin to include explicit discussions of urban poverty and possible responses by computer professionals. The ACM national lecture series that year included Lawrence Gutstein, corporate planning representative at IBM, on "The Computer and the Ghetto: a Means of Social Change."[54] That year's ACM annual symposium on the Application of Computers to the Problems of Urban Society featured a discussion of racial desegregation in schools as well as a screening of the film "Portrait of South Bronx" which was reported to have "griped [sic] each Symposium attendee in the daily emotional experiences of the residents in a typical urban problem area." The annual Urban Symposium, which was called "Metropolitan" rather than urban in its first year, usually focused on issues such as

53. A. G. Oettinger. 1967. President's letter to the ACM membership. *Commun. ACM*, 10(3): 139–140.

54. ACM national lecturers 1968–69. *Commun. ACM*, 1968, 11(10): 718–719.

traffic control, highway planning, and air pollution and rarely addressed, at least explicitly, the concerns and interests of communities of color.

The September 1968 issue carried the first article discussing one of these emergent training programs and the first one developed under the auspices of an ACM chapter. Carrying the headline "Greater Boston Chapter Establishes EDP Training Program for the Underpriveleged" [sic], the article describes a program for instruction in basic data processing, computer operations, and programming developed by the Greater Boston chapter, by way of its own local Special Interest Committee on Social Concern, in cooperation with the Opportunities Industrialization Center (OIC)—identified as "a black self-help group located in the Roxbury area of Boston."[55]

The next appearance in *CACM* of a training program, though not one by an ACM chapter or other affiliate, came in the November 1968 issue. "Los Angeles Chapter Announces Support for Placement of Disadvantaged in EDP Jobs" drew attention to a program in "computer operation, basic programming and key punch techniques" developed through a partnership between the Los Angeles Urban League, Bank of America, and IBM.[56] Janet Toland reveals in her chapter in this volume that ACM SICCAS (as it was first known, having been established as a special interest committee) discussed this program in a panel on the "Education of Non-Computer People" in November 1969 which characterized the program's target audience as the "undereducated."[57] The Los Angeles chapter of the ACM later became involved with this program, when in 1971 they were reported to have volunteered as facilitators for the placement of students who completed the program of what by then had come to be known as the Urban League Training Center.[58] But the November 1968 issue also reported that they hosted a lecture on "The Ghetto and the Computer Pro-

55. Greater Boston chapter establishes EDP training program for the underpriveleged [sic]. *Commun. ACM*, 1968, 11(9): 647.

56. Training center for disadvantaged offers courses in computer operations. *Commun. ACM*, 1968, 11(11): 795.

57. This is the only instance I have discovered to date where the term "undereducated" was used. By far the most common term employed was "disadvantaged." The second most common term is "underprivileged." While each term has differing connotations, all seem to function as code words which obscure racial and ethnic identities.

58. Los Angeles chapter announces support for placement of disadvantaged in EDP jobs. *Commun. ACM*, 1971, 14(1): 58. Toland's source is R. P. Bigelow. 1970. From the chairman. *Comput. Soc.*, 1(2): 2. The report of the SIC's Education Subcommittee, found on p. 10 of the same issue, categorizes four different communities of "non-computer people" whose needs they desired to address: "Disadvantaged," "General Public," "Students," and "Government Officials." The

fessional" by Sam Feingold of SDC and Bob Hall of Operation Bootstrap, described as "a nonprofit organization whose goals are to educate from within, members of the black community."[59] The December 1968 issue closed out the year with a report on the program developed at Johns Hopkins University, mentioned above.

In August 1969 *CACM* described the involvement of the New York city chapter in providing volunteer teachers and textbooks to an "anti-poverty training program." The specific program is not identified, but the description reads, "EDP courses were given in keypunching, console operating, and programming to groups of hard-core unemployed in New York City."[60] Of the three programs operating in New York at that time, this description aligns with that provided by Mayer of the Middle West Side Data Processing School.[61] The program developed by the Computer Personnel Development Association purportedly offered only computer operator training, and the Harlem program at Columbia focused on IBM keypunch operator training.

Two additional programs developed by ACM chapters would next be reported among those discussed at three conferences in 1969. The Delaware Valley chapter's Urban Education Committee offered a computer operator program focused on the IBM 360/30.[62] The Sacramento chapter offered a computer operator program, identified in 1969 as the work of the chapter's Education Committee[63] and later said to be administered by its Special Interest Group on Training and Placement of the Culturally Disadvantaged (SIGTAP), a "special chapter group of chapter members and other concerned computer professionals."[64]

Other programs may not have been established under the aegis of official ACM chapters, but ACM members played a significant role in them. These include two New York-based programs: the Harlem IBM keypunch operator courses conducted

communities listed under "Disadvantaged" include: "Poor," "Blacks," "Indians," "Puerto Ricans," and "Mexican-Americans."

59. Larsen discusses EDP schools at NYC chapter. "Los Angeles," and "Dallas," under "Topics at Chapter Meetings." All appearing in the "News" section *Commun. ACM*, 1968, 11(11): 791–792.

60. New York City chapter elects officers. *Commun. ACM*, 1969, 12(8): 481.

61. D. B. Mayer. 1969. The involved generation: Computing people and the disadvantaged. *AFIPS '69 (Fall) Proceedings of the November 18–20, 1969, Fall Joint Computer Conference*, p. 689. ACM, New York.

62. H. Griffin and P. Gravelle. 1969. Report on the philosophy and mechanics of the Urban Education Committee of Philadelphia. *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*, pp. 95–98. ACM, New York; Mayer, The involved generation.

63. Mayer, The involved generation, 689.

64. School dropouts trained and placed as computer operators by ACM Sacramento chapter. *Commun. ACM*, 1970, 13(10): 641.

at the Columbia University Computing Center by Walter A. DeLegall, and computer operator courses offered by a group identified as the Computer Personnel Development Association (CPDA) which included active ACM members such as Jean P. Gilbert, Allen L. Morton, Jr., and David B. Mayer.[65] One fully in-house corporate program conducted by the MITRE Corporation in Boston included the participation of James H. Burrows. Although the activities of these individuals were not conducted on behalf of ACM or its chapters, they did present on their work at relevant conferences sponsored by ACM's affiliate and member organizations AFIPS and SIGCPR, and some played active leadership roles in ACM and SIGCPR committees interested in working with the disadvantaged.

10.5 ACM Establishes a National Committee on Computing and the Disadvantaged

Although reports of training programs first appeared in *CACM* in 1968 and did not hit the conference programs until 1969, awareness of these activities was sufficient to mobilize members and leaders active in ACM groups, who did not fail to announce their own activities. In the October 1968 issue of *CACM*, ACM President Bernard Galler announced the formation of the Committee on Computing and the Disadvantaged in a President's Letter titled "Involvement."[66] Additionally, "Computers and the Disadvantaged" appeared in the ACM Reference Guide under the heading "Other Committees Which are De Facto Standing." The position of committee chair was listed as "open."[67]

Concurrently, SIGCPR announced plans to develop its own committee. An announcement in the November 1968 issue read:

> The committee is charged with the responsibility for compiling information on existing programs and encouraging research into the many aspects of the problem. The ultimate objective is to provide the computing community with accurate

65. J. P. Gilbert and D. B. Mayer. 1969. Experiences in self-selection of disadvantaged people into a computer operator training program. *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*, pp. 78–90. ACM, New York; V. Henriques. 1970. Computing and the disadvantaged, a panel session. *ACM '70: Proceedings of the 1970 25th Annual Conference on Computers and Crisis: How Computers Are Shaping Our Future*, pp. 8–15. ACM, New York; Mayer, The involved generation; and A. L. Morton, Jr. 1969. Computers and the underprivileged. *AFIPS '69 (Spring) Proceedings of the May 14–16, 1969, Spring Joint Computer Conference*, pp. 37–38. ACM, New York.

66. President's letter to the ACM membership: Involvement. *Commun. ACM*, 1968, 11(10): 659.

67. ACM reference guides: ACM committees and chairmen (as of September 15, 1968). *Commun. ACM*, 1968, 11(10): 726–727.

> and complete data regarding the occupations in which the disadvantaged might be successfully utilized, to establish selection criteria, and to organize training and placement programs specifically designed for the disadvantaged. Training development will be coordinated with the SIG's Standing Committee on Training Methodology.[68]

Acknowledging the existence of multiple programs, the article goes on to state that a primary task of the committee would be to compile a history of such programs. The call for papers for the SIG's seventh annual conference to be held June 19–20, 1969 included as one of its five areas of interest, "Selection Criteria and Training programs for Disadvantaged Individuals for Entry-Level Jobs in the Computer Profession."[69] In December 1968 an article announcing David B. Mayer, a long-time active member of the special interest group, as the new chair of SIGCPR, also stated: "Since current trends have lead SIGCPR in a new direction, a second committee on Research into Entry Level Jobs for Disadvantaged in Computing was established and is headed by Mr. James Burrows of the MITRE Corporation."[70]

In 1969, the newly formed ACM committee took off and appears to have surpassed the efforts of the SIGCPR committee. Actually many of the same individuals were involved in both. Minutes of the ACM Council Meeting on December 12, 1968 stated briefly that James Burrows reported on the plans and activities of the new ACM committee but announced that David Mayer was the chairman.[71] A brief article announced the appointment more prominently in February.[72] A recap of the Fall Joint Computer Conference stated that Mayer, appearing as chair of the new committee, reported to a conference of chapter chairmen on the chapter activities in educating the disadvantaged. Burrows was still active and appears as the session chairman of a panel on "Computers and the Underprivileged" held at the AFIPS Spring Joint Computer Conference,[73] and the June 1969 SIGCPR Conference contained three papers that responded to the call for presentations addressing the

68. SIGCPR will study occupational programs in information processing for disadvantaged. *Commun. ACM*, 1968, 11(11): 793.

69. SIGCPR call for papers. *Commun. ACM*, 1968, 11(11): 797.

70. SIGCPR announces new officers. *Commun. ACM*, 1968, 11(12): 866.

71. ACM council meeting, December 12, 1968: Minutes summary. *Commun. ACM*, 1969, 12(2): 66.

72. David Mayer to chair committee on computing and disadvantaged. *Commun. ACM*, 1969, 12(2): 121.

73. J. H. Burrows. 1969. A panel session: Computers and the underprivileged. *AFIPS '69 (Spring) Proceedings of the May 14–16, 1969, Spring Joint Computer Conference*, pp. 35–40. ACM, New York.

disadvantaged. This amounted to one fourth of the conference program. These included presentations by Walter A. DeLegall, Jean P. Gilbert, and David B. Mayer (co-presenting), and Howard Griffin and Paul Gravelle (co-presenting).[74]

The ACM Committee on Computing and the Disadvantaged appears on the program of the 1969 ACM National Conference with both a closed working session and an open workshop:

> A panel of three experts in the training and/or computing field will direct a discussion into all phases of this most interesting area of involvement for the computing professional. Director of the workshop will be Mrs. Helaine S. Dawson of San Francisco. Mrs. Dawson is the author of the book *The Outskirts of Hope*. She specializes in organizing projects and teaching teachers as well as amateurs how to involve themselves in a meaningful training project for disadvantaged peoples.[75]

An August 1970 *CACM* article "Committee on Computing & Disadvantaged Receives Grant" announced with great fanfare:

> The ACM Committee on Computing and Disadvantaged has received a $6,000 planning grant from the Columbia University Urban Center, which is funded by the Ford Foundation. The grant is specifically earmarked for the planning of a large scale training project within ACM chapters for entry level jobs in computing for the disadvantaged.[76]

The article stated that proposal guidelines had already been drawn up and that Mayer had already begun speaking with ACM chapters "advising on the establishment of a local training project for generating guidelines to schools on directing disadvantaged people into computer training." Furthermore, the committee expected:

> to also generate a proposal for a large scale training system over a five-year period, that would include placement in jobs. This project would emphasize, first, techniques for getting jobs; second, the training; and then placement in the openings for which commitments were obtained.

---

74. W. A. DeLegall. 1969. Teaching techniques and quality education/training for the 'disadvantaged'. *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*, pp. 91—94. ACM, New York; Gilbert and Mayer, Experiences in self-selection of disadvantaged people into a computer operator training program, 79–90; and Griffin and Gravelle, Report on the philosophy and mechanics of the Urban Education Committee of Philadelphia, 95–98.

75. Workshop on computing and disadvantaged. *Commun. ACM*, 1969, 12(8): 422.

76. Committee on computing & disadvantaged receives grant. *Commun. ACM*, 1970, 13(8): 518.

Several people were identified as being already involved, including many mentioned above: Jean Gilbert, Allen Morton, Paul Gravelle, Howard Griffin, Helaine Dawson, and others.

The Committee seemed well poised to continue its work, with the participation and leadership of people who had already demonstrated themselves as capable and committed to such a project and now with the support of a sizeable grant. Mysteriously, no further mention of the Committee's plans or grant appear in *CACM*. Instead, the January 1971 issue of *CACM* bore the terse statement "David Mayer, chairman of the Committee on Computers and the Disadvantaged, has resigned."[77] Little mention is found of Mayer in subsequent issues, though he is found on the roster of the ACM national lecture series for 1972, offering presentations on "Personnel Selection & Training of Disadvantaged."[78]

After Mayer's departure, the Committee appears to go have gone through some transitions. In a letter to ACM President Walter M. Carlson dated April 1, 1971, Milton Bauman accepts the chairmanship of the committee. He writes, "As I understand it, my first task will be to organize a committee to write a proposal for a Massive Training Program for the Disadvantaged in the Computer Sciences." Bauman's chairmanship is announced in the May 1971 issue of *CACM*, under the incorrect heading "Bauman to Head ACM Committee for the Handicapped." It is, in fact, the Committee on Computing and the Disadvantaged that he will lead. No further mention of Bauman appears after this. An August 1971 report on the restricting of committees would place the Committee under the auspices of a new External Activities Board headed by Herbert S. Bright. The same article identifies the committee now as "Ad Hoc" rather than its previous designation under "Other Committees Which are de Facto Standing."

From this point on, the Committee on Computing and the Disadvantaged disappears from the pages of *CACM* except for its ongoing inclusion in the ACM Reference Guides until its last appearance in the January 1974 issue. All further information about the Committee comes from one folder in the Association for Computing Machinery Records.[79] A letter dated September 11, 1972 from ACM President Anthony Ralston to M. Granger Morgan, confirmed the verbal acceptance of co-chairmanship of the Committee on the phone the week prior and identified Morgan's fellow co-chair as Robert Sadler.

77. ACM council meeting, November 19–20, 1970: Minutes summary. *Commun. ACM*, 1971, 14(1): 54.

78. ACM lecturers 1972. *Commun. ACM*, 1972, 15(2): 120–121.

79. Association for Computing Machinery Records, CBI 205, Box 25, Folder 13.

A memorandum filed by Morgan and Sadler, dated March 22, 1973, outlines the proposal of the new co-chairs for use of the grant from Columbia. They state:

> . . . we believe that it would be unwise for the ACM Committee on Computing and the Disadvantaged to pursue its original objective of several years ago. . . . the establishment of one or two major "demonstration" training programs for "disadvantaged students". . . . We propose to organize a small invitational workshop involving about ten people who have been working directly in the field of computers and the disadvantaged, or in closely related fields, with the objective of developing guidelines and a collection of printed and audio-visual material which will encourage and provide support to the individual ACM member or local chapter to become involved in in-school and community based motivational and orientational activities or in career information and reference services directed at the disadvantaged.

The letter includes rationale for this decision, specific goals and anticipated outcomes for the new proposal, and a proposed budget. The entire package bears a cover letter from J.M. Adams dated August 14, 1973, stating that the enclosed proposal would be voted on by the External Activities Board at its meeting two weeks later on August 28.

Nothing further on the proposal or the vote appears in the file on the Committee itself, but additional documentation is found within the files of the External Activities Board. In the March 1974 annual Committee report, Sadler wrote that the ACM Executive Committee had denied a recommendation on funding. No details or explanation is provided. Under the heading "Problems" he wrote simply, "No money." Another letter from the Committee files dated December 13, 1973 from Morgan to Ralston announces Morgan's resignation. In the letter he states:

> Because the expected support for the committee's proposed activities has not materialized the committee is faced with the task of seeking support from outside sources. I think this type of activity is inappropriate for me as an NSF staff member.

Ralston's response accepting the resignation is also extant. The only other document in this file is a letter dated July 1, 1974, from then ACM President Jean Sammet to Robert Sadler confirming termination of the Ad Hoc Committee in a conversation on June 26, 1974. In it, Sammet writes:

> I really regret that it has been impossible so far for you to find the funding to carry off this type of activity, because I believe it is important for ACM to be doing things like this. Unfortunately ACM cannot provide funds from its own resources either now or in the foreseeable future.

The quiet termination of the Committee suggests an untold story. It was not uncommon for committees, special interest groups and committees, and even chapters to be decommissioned on the basis of a lack of participation or an insufficient number of members, and such changes were commonly announced in the pages of the *CACM*, often with articles announcing the initiation of procedures to terminate a group and allowing for action to be undertaken to prevent it through the accumulation of signatures on a petition. No such action was publicized concerning the ACM's brief efforts to aid disadvantaged, urban communities of color.

10.6 Conclusion

The ACM has a history of engaging communities of color in computing, although such efforts were short-lived and limited to communities defined as "disadvantaged." Such programs were created largely due to the altruistic motives of a segment of the computing community who desired to find solutions to joblessness and poverty. It was, however, the support of government agencies and business that allowed these programs to persist for as long as they did. While some individuals may have been focused on the needs and desires of the communities themselves, as David B. Mayer put it, "to appear at that interface between their own computing community and those underprivileged who wish to enter it," others were motivated by different interests. In a turbulent era marked by civil unrest, socially motivated individuals may have found themselves joined by others urged on by the sense of needing to avert a crisis. Businesses had the added incentive of government initiatives and their own internal need to meet the demand for more personnel trained in computing functions while meeting their own goals for minimizing costs.

James P. Titus wrote in ACM's Washington Commentary of August, 1968, that "Boards, committees, and commissions rise and fall with the fiscal tides of Washington."[80] One could say that educational efforts aimed at helping underserved and underrepresented communities likewise rose and fell with government and business trends. As these activities took place amidst a national climate that emphasized social awareness and involvement on a host of issues, including military actions with which the computer industry was also involved, it appears that motivated computer professionals were able to ride the crest of a wave of awareness and activity. Once the need to avert a "crisis" had abated, they similarly fell as support was withdrawn.

80. Washington commentary: The new NAS board as government advisor. *Commun. ACM*, 1968, 11(8): 580–581.

Of course this is too simple a picture, and other factors were undoubtedly responsible, including rising expectations for computer workers at even entry levels and the role of private EDP schools, whose own life cycle was not examined here. What remains unclear is the perspective and experience of the students, both during and after training. While the programs themselves may not have continued, we do not know if these individuals found in these programs a viable entry point into a career that allowed them to grow and succeed. Lessons learned from these programs in regards to student success and program sustainability could very well be of immense value to current efforts in STEM education and diversity in computing.

11 Other Places of Invention: Computer Graphics at the University of Utah

Jacob Gaboury

11.1 Introduction

The history of computing in the U.S. is dominated by government-funded universities and private research institutions on the East and West coasts. MIT, Stanford, Harvard, UCLA, UC Berkeley, Bell Labs, Xerox PARC, and others played key roles in the early development of computer science and the transformation of that field into a large-scale industry. Often overlooked are those sites and institutions outside the research meccas of Silicon Valley and Boston's Route 128, spaces whose fundamental contributions to the history of computing have been largely forgotten, even by the institutions themselves.[1] The University of Utah is precisely this kind of place. Not only a major computer science research center of its time, it was also the premier institution for research into computer graphics in the United States for over 15 years. During roughly 1966–1979 the faculty and graduates of the Utah program were responsible for no less than inventing the concepts that make modern

1. In 2006 Professor H. Kent Bowen and Courtney Purrington, Ph.D., were commissioned to prepare a Harvard Business School case study on the University of Utah phenomenon by recently elected university president Michael Young. In 2004 Young learned of Utah's prestigious graphics history at the dedication of the new Warnock Engineering Building, named for Adobe Systems founder John Warnock, at which time there seemed very little institutional memory of the university's significant contributions. I draw on Bowen and Purrington's work in what follows, with particular interest to their original interviews with Utah faculty and alumni. See H. K. Bowen and C. Purrington. 2006. The University of Utah and the computer graphics revolution. Harvard Business School, Unpublished Case Study, April 28, 2006; revised April 11, 2007.

computer graphics possible. Some of the very first experiments with raster graphics, frame buffers, graphical databases, hidden surface removal, texture mapping, object shading, and more were conducted in the graphics lab in Salt Lake City, and many of the school's graduates went on to become industry leaders in the field. The Utah program served, in effect, as a testing ground for computer graphics in the decade leading up to the dramatic expansion of the Association for Computing Machinery's Special Interest Group on Graphics and Interactive Techniques (ACM SIGGRAPH) and the growth of what would eventually become the modern computer graphics industry.[2]

The Utah program's success was due in large part to the enthusiasm and dedication of a small handful of researchers in this very young field, and the faith and financial backing of several directors at the Advanced Research Projects Agency's Information Processing Techniques Office. Chief among these individuals was David C. Evans, a Utah native who was the founder and driving force behind the department, and who served as mentor to dozens of key figures. While Evans is by no means the only important figure in this history, the trajectory of influence that he holds over so many major players and developments in the field of computer graphics speaks to the unique environment he was responsible for cultivating at Utah in the 1960s and 1970s. At minimum it speaks to the culture of mentorship he fostered, which may be traced from these early moments to the "culture of innovation" promoted by so many Silicon Valley startups to come after it. The founders of Pixar, Adobe Systems, Silicon Graphics, Atari, Netscape, and WordPerfect were all students at Utah during this critical period. Likewise, key research centers at Xerox PARC, NASA, LucasArts, Pixar, and elsewhere were founded and populated by Utah alumni. The influence of the University of Utah program on the contemporary field of computing is massive, and is threaded throughout the field of computer science and its transformation into a commercial industry, yet it has until recently been largely neglected as a site for critical investigation.[3]

2. J. Brown and S. Cunningham. 2007. A history of ACM SIGGRAPH. *Commun. ACM*, 50(5): 54–61.

3. To be sure, the history of computer graphics does not begin with Utah. The development of early two-dimensional computer graphics finds its roots in the SAGE air defense system and with research teams at MIT, Lincoln Lab, and Harvard. Likewise, a great deal of early experimentation was conducted by artist-researchers such John and James Whitney, who explored rudimentary computer visualization through oscillography and early analog computing technology. See J. Hurst, M. S. Mahoney, N. H. Taylor, D. T. Ross, and R. M. Fano. 1989. Retrospectives I: The early years in computer graphics at MIT, Lincoln Lab and Harvard, and Retrospectives II: The early years in computer graphics at MIT, Lincoln Lab and Harvard. *SIGGRAPH '89 Panel Proceedings*, July 31–August 4, 1989, pp. 19–73. Association for Computing Machinery, Boston/New York; Z. Patterson.

Utah's peripheral status in the history of computing in many ways reflects the field of computer graphics as a whole, which has been dominated by these secondary sites and non-traditional institutions. This is due in part to the intersection of both research and artistic interests in this uniquely visual field, along with the broad trajectory that computer graphics would take over the course of its history—from the development of industrial applications for computer-aided design at General Motors to rendering software and computer animation at film studios such as Pixar, and contemporary applications in graphical user interface design, digital gaming, and data visualization. This is also due to the experimental nature of the work itself, and the fact that it was not until several decades of financial and intellectual investment that computer graphics became deployable outside of heavily funded university or industry contexts. Graphical visualization had long been a trope of early computing, particularly as it was popularized in other visual media such as film and television, but the computational requirements of visual computing were hugely demanding, making most early systems entirely unfeasible for large-scale deployment.[4]

The Utah program sought to change that, and was supported by a small network of researchers and funding entities that would make graphical man-machine communication—along with time-sharing and artificial intelligence—one of the primary research goals for the 1960s. While it would be more than two decades before interactive computer graphics would become an everyday reality for the vast majority of computer users, looking back there is little doubt that the Utah graphics program was a success. For better or worse, contemporary computing for the majority of users is effectively synonymous with computer graphics, as graphical interaction is the primary means by which we engage with computing today. What may be less clear is why the environment at Utah fostered such a brief but frenzied period of innovation, and what effect it may have had on the development of this emerging field.

This chapter asks why the University of Utah was so successful in its push to implement and commercialize computer graphics in this brief 15-year period.

Summer 2009. From the gun controller to the mandala: The cybernetic cinema of John and James Whitney. *Grey Room*, 36: 36–57.

4. The SAGE system required a massive four story building filled with millions of dollars of computing technology and manned by dozens of trained professionals in order to function, and even then there are doubts as to its ability to handle a "hot war" situation, if one had arisen; see P. N. Edwards. 1996. *The Closed World: Computers and the Politics of Discourse in Cold War America*, pp. 104, 110. MIT Press, Cambridge, MA.

I draw on a wealth of archival materials, with principal focus on the David C. Evans Collection of the University of Utah in Salt Lake City. The Evans papers are substantial and cover material related both to the Utah graphics program and its subsequent commercialization through the Evans & Sutherland Computer Corporation. Along with the Evans papers I have drawn from the papers of Thomas Stockham, who replaced David Evans as head of the Utah program in the 1970s, as well as the papers of James Fletcher, president of the University of Utah throughout much of the 1960s. I have also drawn on the Computer Science Department's own archival records, which include images of the graphics lab along with technical reports relating to the University's ARPA contracts and other research.[5] These are supplemented by oral histories from key researchers including David Evans, Ivan Sutherland, and Robert Taylor, as well as interviews with various Utah graduates. Finally, I have drawn on the ACM SIGGRAPH collections housed at University of Minnesota's Charles Babbage Institute to trace the institutionalization of graphical research, and to see the ways in which Utah research began to circulate within the emerging graphics industry.

In 2008, Jeffrey Yost, writing as the editor of the *IEEE Annals of the History of Computing*, clearly noted that "the history of computer graphics is an important topic that has been understudied, particularly for certain regions," and called on researchers to engage with the fifty year history of the discipline. To date, however, few have taken up this task. While recent years have seen a renewed interest in the Utah graphics program, both among historians of computing and by the University of Utah itself, there has been relatively little published on this history and its significance to the broader history of computing. The most significant contribution is Tom Sito's *Moving Innovation: A History of Computer Animation* (2013), which offers a valuable survey of computer animation in the United States from roughly 1950 to the present.[6] However, the book's breadth comes at the cost of attention to any single topic in great detail, and the history of the Utah program takes up little more

5. While less visible in this chapter, this work has been supplemented by the trade collections of the Smithsonian's National Museum of American History, as well as the Link Simulator Company's collection at the Smithsonian's National Air and Space Museum. I have also drawn on the Kurt Akeley Papers and the Russell Vernon Anderson Papers at the Stanford University Archives, as well as the collections of the Computer History Museum in Mountain View, CA. Finally, I have drawn on the Ed Catmull papers and several original oral histories housed at Pixar historical collections in Oakland, CA, and supplemented this work with my own original interviews with Alan Kay, Alvy Ray Smith, Richard Riesenfeld, and Al Davis.

6. J. Yost. 2008. From the Editor's desk. *IEEE Ann. Hist. Comput.*, 30(1): 2–3; T. Sito. 2013. *Moving Innovation: A History of Computer Animation*. MIT Press, Cambridge, MA.

than five pages. Likewise, while substantial monograph-length treatments of artistic experimentation with computer graphics during this early period have become quite common,[7] little attention has been paid to the significant contributions of the Utah program, which was largely responsible for establishing the field of computer graphics as an academic discipline and commercial technology prior to the founding of a dedicated special interest group through the ACM.

In what follows I outline the formation and first decade of the Utah graphics program, framing the kinds of social and professional networks that made the program possible alongside the institutional support that allowed it to thrive. I then examine the culture of mentorship that was fostered by the graphics program with particular focus on two of its lead faculty, David Evans and Ivan Sutherland. Drawing on an important trope of early graphics research known as the "ten unsolved problems," I look to frame the concerns of the then-nascent field of computer graphics while also suggesting that a future-oriented culture of problem solving helped bolster the Utah program and allowed for the rapid expansion that marks this early period of research. I conclude by discussing the program's afterlives and the large-scale shift away from government funded university centers and toward an emerging graphics industry, facilitated by the professionalization of computer graphics through the ACM and its SIGGRAPH special interest group. In doing so I reflect on the role of secondary sites in the history of computer science more broadly, and on computer graphics as a neglected but critical field for almost all contemporary computational applications.

11.2 Salt Lake City, 1966

Why, in 1966, did Salt Lake City become the epicenter for cutting-edge research into computer graphics? The answer is in some ways very simple. It was David Evans' decision to return to Salt Lake City—the city in which he was born and the home of his family and Mormon faith—that set in motion decades of innovative research into graphical applications in computing technology, along with the careers of dozens of key researchers in the field of computer science. In this sense the key role that Utah plays in this history is due largely to chance, or at the very least the

7. See P. Brown, C. Gere, N. Lambert, and C. Mason. 2009. *White Heat Cold Logic: Early British Computer Art 1960–1980*. MIT Press, Cambridge, MA; H. Higgins and D. Kahn, editors. 2012. *Mainframe Experimentalism: Early Computing and the Foundations of the Digital Arts*. University of California Press, Berkeley; and M. Rosen, editor. *A Little-known Story about a Movement, a Magazine, and the Computer's Arrival in Art: New Tendencies and Bit International, 1961–1973*. MIT Press, Cambridge, MA.

coincidence of birth and the belief in a kind of divine providence that makes Salt Lake City the holy center of the Mormon faith. The more difficult question, then, is how did this happen? What were the conditions that made possible this success, the structures that supported this work financially, culturally, and technologically? Put simply, how did the University of Utah as an institution come to support this highly experimental field of research? To answer this question we must begin with David Evans, as it is Evans' experience in the field of computing and his connections with government, academia, and industry that made this history possible.

Born in Salt Lake City in 1924, David Evans received his undergraduate degree in physics from the University of Utah in 1949 after interrupting his studies to serve in WWII.[8] He spent much of the following decade in California as a senior physicist at the Bendix Corporation in Los Angeles. At this early stage in the 1950s computing was a relatively small industry, with computer hardware often developed alongside other industrial electronics as part of contracts for the military or avionics industries. As such, Bendix was not exclusively a computing company, but in 1955 Evans was promoted to director of engineering of the Computer Division, which gave him the responsibility for research, development, and product design for commercial computing systems and special purpose information processing systems for military and industrial applications. While at Bendix, Evans was able to direct two of the company's most noteworthy projects: the Bendix G-15, introduced in 1955 and one of the first inexpensive general-purpose computers to be mass-produced; and the Bendix G-20, introduced in 1961. The chief designer for the Bendix G-15 was Harry Huskey, who had worked part-time on the early ENIAC computer in 1945, and later on the Pilot ACE at the National Physical Laboratory with Alan Turing. Huskey developed much of his research for the Bendix G-15 while a faculty member at UC Berkeley, and in 1962 Evans would follow him there, taking up a tenure-track position and leaving Bendix shortly before its computer division was taken over by the Control Data Corporation in 1963. It was during this period that he began working on projects funded by the Defense Department, serving as co-principal investigator (PI) for a four-year $1.5 million research initiative on computer-aided problem solving funded by the Advanced Research Project's Agency's Information Processing Techniques Office (ARPA IPTO).[9] By 1964 the project had evolved into the Project

8. Transcript, David C. Evans Oral History Interview, April 18, 1996, by Daniel Morrow, p. 11. Computerworld Honors International Archives.

9. David C. Evans (Curriculum Vitae), Box 18, Folder 10, Coll. 0199, James C. Fletcher Presidential Records, 1937–1971, University of Utah. For background on IPTO and graphics, see A. L. Norberg

Figure 11.1 **Bendix G-15 computer (designed by Alan Turing's colleague Harry Huskey) formed a link between Evans early work at Bendix and UC-Berkeley computer science.** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

GENIE time-sharing initiative, one of the earliest time-sharing systems ever built, and the earliest useful realization of time-sharing on a minicomputer.[10]

and J. E. O'Neill. 1996. *Transforming Computer Technology: Information Processing for the Pentagon, 1962–1986*, Chapter 3. Johns Hopkins University Press, Baltimore.

10. See L. P. Deutsch and B. W. Lampson. April 1965. SDS 930 time-sharing system preliminary reference manual, Doc. 30.10.10, Project GENIE, University of California at Berkeley; D. M. Ritchie and K. Thompson. 1974. The UNIX time-sharing system. *Commun. ACM*, 17(7): 365–375; M. A. Hiltzik. 1999. *Dealers of Lightning: Xerox PARC and the Dawn of the Computer Age*, pp. 18–19. Harper Collins, New York; P. Spinrad and P. Meagher. Project Genie: Berkeley's piece of the computer revolution. Available at http://coe.berkeley.edu/news-center/publications/forefront/archive/forefront-fall-2007/features/berkeley2019s-piece-of-the-computer-revolution (accessed November 25, 2012).

It was also during this period that Evans began working with Ivan Sutherland, who in 1964 at the age of 26 had been made head of the IPTO and entrusted with a $15 million budget to continue and extend J.C.R. Licklider's vision of man-machine communication.[11] It was Sutherland who served as Evans' research sponsor at ARPA, and whose vision of *graphical* man-machine communication would have a substantial influence on the then-emerging field of computer graphics (indeed, Sutherland received the 1988 ACM Turing Award for "pioneering and visionary contributions to computer graphics"). While work on the time-sharing project at Berkeley was much smaller in scope than similar ITPO funded research sites of that time—most notably Project MAC at MIT—it produced a network of individuals with lasting ties to the IPTO. As Sutherland recalls, Project GENIE provided a training ground for researchers who would later make significant contributions to the field of graphics at Utah and elsewhere.[12] After two months as co-PI for Project GENIE Evans handed control over to Wayne Lichtenberger—a new visiting assistant professor at Berkeley—as his interests began to shift from time-sharing to graphics.[13]

It was in this context that Evans was approached by University of Utah president James Fletcher to return to his alma mater in Salt Lake City and found a computer science division within the School of Engineering.[14] By that time in 1965 Fletcher was a well-known research engineer in the aerospace industry, and had previously served as faculty at both Harvard and Princeton.[15] He had been appointed president of the University of Utah only one year prior in 1964, and one of his first moves was to recruit Evans. Fletcher's offer came with the full backing of the university, appointing him Director of Computer Science and Computer Operations at the University of Utah—a title intended "to indicate that [Evans would be] in charge of

11. Ivan Sutherland Oral History, Charles Babbage Institute, OH 171, p. 26. Available at http://purl.umn.edu/107642.

12. Ibid., 29.

13. Evans would return to time-sharing again in 1969 when Utah became one of the first four hubs in the ARPANET.

14. Hiring recommendation from U of U computer committee to James Fletcher, May 18, 1965, Box 18, Folder 11, Coll. 0199, James C. Fletcher Presidential Records, 1937–1971, University of Utah.

15. James Fletcher received his Ph.D. in Physics from CalTech in 1948. He was the son of the eminent Bell Labs scientist Harvey Fletcher, known as the "father of stereophonic sound" and credited with inventing the hearing aid. After serving at Utah from 1964–1971 Fletcher would go on to become the head of NASA from 1971–1977 and again for three years following the Challenger Space Shuttle disaster in 1986.

all aspects of computer work . . . at the university."[16] While university funds may have been limited, Fletcher's generous offer was supplemented by a $5 million grant from the IPTO that Evans was able to secure immediately following his hire. Paid out over the course of four years, the ARPA contract was devoted explicitly to "Graphical Man/Machine Communication" and was facilitated initially by IPTO director Ivan Sutherland who would leave government work that same year to take up a tenured position at Harvard University.[17] In the spring of 1965 Evans accepted the position at Utah, moving home to Salt Lake City in 1966 where he would remain for the rest of his life.[18]

To be clear, Evans is but one significant figure in this history. Nonetheless if we wish to understand why and how the Utah program was so instrumental to the development of early computer graphics, we must understand the central role Evans played in its formation and construction. Moreover, unlike many of the key players in this history who went on to found companies that have become household names over the past fifty years, the legacy of David Evans in the history of computing is little known outside of Salt Lake City, despite his central role in the shaping of our modern computing landscape. As this chapter will show, it is not possible to reconstruct the history of the Utah program through the biography of a single individual, as a range of historical actors both institutional and interpersonal played significant roles in producing the culture of research that made Utah so successful. Nonetheless there are key figures on which critical moments hinge, and whose vision shaped the field of computer graphics.

11.3 Practical Applications

In the mid-1960s few universities had programs let alone departments devoted to computational research.[19] Those programs that did exist were usually embedded within a larger primary department, the focus of which often dictated the research

16. Letter from Vice President J. A. Adamson to David C. Evans, June 16,1965, Box 18, Folder 12, Coll. 0199, James C. Fletcher Presidential Records, 1937–1971, University of Utah.

17. Ivan Sutherland Oral History, 11.

18. By all accounts Evans was happy at Berkeley, and prior to Fletcher's offer had no intention of leaving. In an oral history with the Evans family, Dave Evans' wife Joy Evans suggests that they expected to be in California the rest of their lives, but when Fletcher made the offer to Evans, the family decided to "come home." Transcript. April 18, 1996. David Evans Oral History Interview, 31.

19. The computer science program at Utah was the 14th degree-granting program in the United States.

direction that the computer science division would take. Frequently this would be a department of mathematics, the discipline out of which the theory of computing first emerged and where theoretical models for computation were actively researched. More practical applications in sub-fields such as computer graphics were often developed through corporate research centers such as AT&T Bell Laboratories[20] or General Motors Research Laboratories,[21] with clear ties to industry-based applications for computer interaction. It is therefore significant that the program at Utah was founded within the School of Engineering, and that its focus was on hardware applications over what Evans described as a tendency toward "mathematical models and abstract investigation."[22] This departmental focus reflected a culture of practice-based applications at Utah that attracted graduate students and faculty interested in making computer graphics deployable not only on a highly funded academic and military scale, but also for commercial use by corporations and individuals.[23] This is also one possible explanation as to why so many of the department's graduates and faculty would leave academia in the 1970s to enter the private sector, often commercializing the very technologies they had developed as IPTO-funded researchers at the University of Utah.

This emphasis on "real computing systems" speaks in many ways to the funding mission of the IPTO at this time. On paper its goal was the long-term development of technologies through the funding of creative research at key institutions, though at this early stage grants were given out largely as "visionary money" to key figures in this burgeoning field with no explicit result in mind. Instead, the program was driven by the creative vision of its director, J.C.R. Licklider, whose model for man-computer symbiosis served as the rallying cry for much ITPO-funded computational research throughout the 1960s, regardless of its potential application to any explicit form of military defense.[24] As Licklider recalled in 1988:

20. Z. Patterson. 2015. *Peripheral Vision: Bell Labs, the SC 4020, and the Origins of Computer Art*. MIT Press, Cambridge, MA.

21. F. N. Krull. 1994. The origin of computer graphics within General Motors. *IEEE Ann. Hist. Comput.* 16(3): 40–56.

22. P. Stewart. May 1968. Faculty profile: David C. Evans. Utechnic, University of Utah Engineering Publication, 10.

23. Indeed one of the origins of *personal* computing is arguably Alan Kay's "The Reactive Engine," which proposed a high-level kernel language called FLEX along with a "personal, reactive, minicomputer which communicates in text and pictures by means of keyboard, line-drawing CRT, and tablet" (vii). See A. Kay. 1969. The reactive engine. Dissertaton, University of Utah.

24. J. C. R. Licklider. 1960. Man-computer symbiosis. *IRE Trans. Hum. Fact. in Electron.*, 1: 4–11.

> the problems of command and control were essentially problems of man computer interaction [. . . so] why didn't we really develop an interactive computing? If the Defense Department's need for that was to provide an underpinning for command and control, fine. But it was probably necessary in intelligence and other parts of the military too. So, we essentially found that there was a great consonance of interest here, despite the fact that we were using different terms we were talking about the same thing.[25]

Under Licklider's plan there was no need to justify funding in terms of military outcomes, although there was a push for practical solutions to the problem of man-computer interaction in the long term. Unlike contemporary models used by funding bodies such as the National Science Foundation (NSF), one of the defining features of the IPTO under Licklider and his immediate successors was the *absence* of a peer-review system. The office functioned instead through a system of informal networks, whereby researchers with a proven ability to run large-scale projects that were known to IPTO directors were awarded large grants to pursue visionary research. It is through this network that Evans was able to secure the $5 million grant to pursue research into computer graphics at Utah.[26]

As Thierry Bardini has noted in his work on Douglas Engelbart and the Stanford Research Institute, this system created a network of insiders who had been graced with the approval of the IPTO, and who collectively shaped the direction of computational research in the U.S. for decades.[27] While Bardini is critical of this insider system for the way in which it excluded Engelbart's work on personal computing in favor of Licklider's vision of artificial intelligence, the Utah program would benefit greatly from this hands-off approach, which by many accounts fostered a culture of research that operated largely independent of any broader consensus of what an appropriate object for computational research might be.[28] As Gianna Walker

25. J. C. R. Licklider, Charles Babbage Institute, OH 150, pp. 24–25. Available at http://purl.umn.edu/107436.

26. It was also through this network, in part, that IPTO directors Ivan Sutherland and Robert Taylor would become University of Utah faculty in the years following their IPTO tenures.

27. T. Bardini. 2000. *Bootstrapping: Douglas Engelbart, Coevolution, and the Origins of Personal Computing*, p. 22. Stanford University Press, Stanford.

28. This hands-off approach is perhaps best exemplified by the incredibly brief semiannual reports written by Evans for the IPTO from 1966–1970. While the reports grow in length and complexity as the program expands, many are shorter than five pages long, offering little in the way of justification, favoring instead brief descriptions of hardware acquisitions and program goals. See D. Evans. Graphical man/machine communications. University of Utah, Semiannual Progress Report for period ending November 30, 1966. ARPA Contract AF30(602)-4277.

recalled in her brief history of the Utah graphics program at SIGGRAPH in 1994:

> They knew that they were "onto something big" while outsiders at other universities disparaged the work in computer graphics as an illegitimate application of computing machinery. Computing research at that time involved computer languages, operating systems, and data processing. Graphics research required manipulating so much data to display images, that it pushed the envelope in computing technology.[29]

Computer graphics research was impractical and unrealistic at this early stage. The technologies did not yet exist, and the computers themselves were not powerful enough to manipulate the massive amounts of data required for interactive graphical communication between a computer and its users. Yet despite these challenges IPTO directors viewed graphical interaction as central to the future of the field, and Evans was tasked with developing the technologies to make these systems possible. In this sense the isolation that Salt Lake City afforded was in fact beneficial to early research, as it allowed for a critical distance and sharp focus on what at that time may have seemed to many researchers outside the IPTO as an improper field of inquiry.

The majority of Licklider's early funding initiatives were in the field of artificial intelligence at institutions such as Stanford, Berkeley, and MIT. It was his successor, Ivan Sutherland, who organized an explicit program to fund graphics research when he took over as director of the IPTO in 1964. By 1965, Sutherland had implemented the Graphic Control and Display of Computer Processes program, whose stated goal was:

> to couple the advanced computer drawing techniques now available with complex internal computation programs. Such coupling will make it possible to control complex computation by drawing and display the results of the computation as drawings . . . The aim is to build tools necessary for controlling a wide variety of computations graphically.[30]

At that time Sutherland was arguably the most prominent researcher in the field of computer graphics, due largely to his doctoral dissertation at MIT, completed only one year prior to his appointment at IPTO. Titled *Sketchpad: A Man-Machine Graphical Communication System* (1963), it is arguably one of the most influential

29. G. Walker. 1994. The inception of computer graphics at the University of Utah 1960s–1970s. (notes from panel at the annual meeting of the Association of Computing Machinery's SIGGRAPH, September 27, 1994). Available at http://archive.today/lADGG.

30. Sutherland quoted in Norberg and O'Neill, *Transforming Computer Technology*, 128.

documents ever written by an individual for its vision of how a user could interact with a computer graphically.[31] For Sutherland visual images were essential tools for understanding complex computational problems, and the funding initiatives of the IPTO during his directorship from 1964–1966 reflect this strong interest. Central to this vision was the funding of the Utah program both by Sutherland and his successor Robert Taylor, who pushed Evans to build a Center of Excellence for computer graphics research at Utah, making it the flagship program for computer graphics research in the U.S.[32] The notion of a Center of Excellence was a key part of ARPA's funding model at this time. First implemented in its Material Science Program, ARPA would find individuals with strengths in a particular field and funnel research funds into that program in the hopes of creating a hub of expertise and collaboration.[33] In effect IPTO would invest money in a field through a single institution and often by way of an individual scholar. If a particular institution failed to coalesce as a Center of Excellence it would be defunded and resources could be allocated elsewhere. This was the case with ARPA's efforts at the University of Michigan to fund a major computer graphics project called Conversational Use of Computers (CONCOMP). At Michigan the objective was to provide theory and programs for a general network service system which, through the use of computer graphics in a conversational mode, would allow researchers to specify specialized graphical problem-oriented language systems for the description and solution of diverse network problems. While the CONCOMP program continued until 1970, the University of Michigan did not evolve into an IPTO center of excellence, and as such its graphics program was not considered a success; it was defunded by the IPTO in 1968.[34]

By funneling money and resources in this focused way, the IPTO was able to concentrate efforts in a particular field and create long-lasting collaborations and connections. While the concentration of talent in particular universities was an asset for focused collaboration, it also created challenges for collaboration between ARPA-funded programs. This was, in part, the justification for ARPA's interest in time-sharing technology, and the funding and development of the ARPANET beginning in 1969 under Robert Taylor. If each research center functioned as a hub,

31. A. Blackwell and K. Rodden. 2003/1963. Preface to this electronic edition. In *Sketchpad: A Man-Machine Graphical Communication System*. Cambridge University Online, Cambridge.

32. Robert Taylor Oral History Interview, October 10–11, 2008, by Paul McJones, p. 16, Computer History Museum.

33. Ibid.

34. Norberg and O'Neill, *Transforming Computer Technology*, 128.

networking those hubs could tap into the expertise and technical resources of the others.

Here we begin to see the institutional infrastructure that made the Utah program possible. It was Evans' experience with the IPTO at Berkeley and his connections to individuals and funding bodies that put him in the position to take on a field of research that was viewed by others as impractical or illegitimate. This institutional network was further enhanced by the administrative framework of the IPTO, in which full trust was granted to Evans and little oversight given as to the direction of his work. This isolation was mirrored in the remote location of Salt Lake City itself. Removed but not inaccessible, this geographic and institutional context allowed for a robust Center of Excellence to grow that would attract dozens of key researchers from around the world who would build a unique culture of innovation over the next fifteen years. How then did Evans and his colleagues take on this task of building a field of research where no framework existed? Where does research begin when no model exists for the objects at hand?

11.4 Problem Solving

While it was Evans' job to build a computer science department that would serve the general needs of the University, his primary focus from the very beginning was research into computer graphics. As Evans described it:

> One interesting thing about starting a new department at a university is there's nothing going on when you get there. There's no equipment and no students and no money, and so you can do one thing as easily as you can do anything else. And so that's when I decided that I ought to be a computer graphics expert, and so I declared myself one and we started.[35]

This narrow focus was a tactical decision. Rather than attempt to compete with other computer science programs at leading research institutions, Utah would focus most all its efforts on one thing only: computer graphics.[36]

35. History of Evans & Sutherland part 1. Audio Recording, June 17, 1977, 07:14–07:50 David C. Evans audio-visual collection A0159, Special Collections and Archives, University of Utah.

36. While graphics was the department's focus, initially there were four specific areas of research: the Computer-Aided Architectural Design System project, headed by Stephen L. Macdonald, from the department of Architecture; the Left-Ventricular Dynamics Project headed by Dr. Homer K. Warner from the Department of Biophysics, a group focused on "The Use of Graphics in the Solution of Partial-Differential Equations" headed by the director of the Computer Center Louis A. Schmittroth, and David Evans' Computer Graphics Techniques division. See D. Evans. May 1968. Computer science at the U of U. Utechnic, University of Utah Engineering Publication, pp. 12–13.

Evans' initial focus at Utah was establishing the computer science division within the School of Engineering, and from 1966–1967 he used his IPTO funding to set up a Computer Center designed to serve the large scale computational needs of the entire campus. Within this Center Evans installed a laboratory built specifically for research into graphical applications. Initially, the Computer Center was set up with an IBM 7044, an early transistor computer released in 1963 but quickly made obsolete by the IBM/System 360 family in 1964. The Center soon replaced the IBM with a UNIVAC 1108, which utilized integrated circuits and allowed for simultaneous handling of real-time applications, time-sharing, and background batch work. This main computer system was then connected to the Graphics Laboratory through a PDP-8 minicomputer that was modified to provide both extended interrupt logic and the analog output required to drive the half-tone cathode ray tube. The PDP-8 was then connected to two graphics facilities, or workstations.

The first workstation was called the Line Graphics Terminal and was composed of a 21-inch cathode ray display with a light pen. The terminal was driven by a line generator and a 64-character symbol generator constructed by Information Displays Incorporated.[37] As its name implies, this terminal was used primarily for simple vector graphics composed of wire frames with no surfaces or shading. This kind of graphical output was relatively simple and had been possible since the early 1960s, but prior to this period it was extremely difficult to produce realistic images with hidden lines removed. The second workstation was the Half-tone Graphics Terminal, which consisted of a modified laboratory oscillograph driven through a digital-to-analog converter by the PDP-8. This terminal would be used for more general-purpose computer graphics services such as line drawings and symbolic displays with light-pen pointing and tracking. The half-tone terminal had the initial capability of displaying half-tone representations of two-dimensional projections of three-dimensional objects generated by the Univac 1108 and PDP-8 computers. This was the primary workstation for early graphics research, and was used in the efforts to solve key problems in representing three-dimensional graphical objects.

The primary objective of the department at this early moment was "to develop a means to produce photograph-like pictures of complex illuminated objects at a reasonable cost."[38] As Evans described in his first ARPA report:

> A major objective of the project is to develop a method which will produce synthetic video signals representing two-dimensional projections of three-

37. D. Evans. 1966. Graphical man/machine communications. University of Utah, Semiannual Progress Report for period ending November 30, 1966. ARPA Contract AF30(602)-4277, p. 2.

38. Evans, Computer science at the U of U, 13.

> dimensional objects described only in the data structure of the computer. It is considered that such a representation of the object is of basic value, because it corresponds to the representation of the real world on the human retina.[39]

What may sound simple enough by contemporary standards was a massive undertaking at the time, and presented a number of challenges that had never before been addressed by computer science. Evans outlined these challenges explicitly in an early report on the computer science department published in the University of Utah's own engineering publication *Utechnic*:

> In order to do this, three important problems must be solved: 1) A good means must be found so that only visible surfaces are displayed in the picture. 2) Halftone shading determined by instant light must be provided. More information beyond that provided by line drawings is given by this means. 3) A good model must be found for representing arbitrarily shaped surfaces.[40]

This method of formulating research goals as problems in need of solution was an important trope for early computer graphics, as it helped to focus and organize this otherwise disparate field of research. By positing technological development and change in terms of actionable problems, researchers created a motivational structure in which otherwise abstract goals such as graphical development or visual mimesis were understood as identifiable problems that were always progressing toward a more accurate or realistic solution, even as Evans' initial goal of true photorealism remains a receding horizon to this day.

Evans was not alone in this problem-solving formulation. Indeed this is a central concept for the modern sciences, and arguably the primary interest of the various branches of engineering, concerned as they are with the design, analysis, and development of technological solutions. Nonetheless, the trope of unsolved problems takes on a particular valence in computer graphics to become a kind of call to arms, a means of mobilizing a disparate group of researchers to a clearly defined set of tasks. This formula begins—as with so many things in computer graphics—with Ivan Sutherland and a brief six-page article written for the journal *Datamation* in 1966.[41] Titled "Computer Graphics: Ten Unsolved Problems," the article became a critical framework for computer graphics researchers for over fifty years, revis-

39. Evans, Graphical man/machine communication, 3.

40. Evans, Computer science at the U of U, 13.

41. The article is often incorrectly cited as "Ten unsolved problems in computer graphics," which has since become the standard formatting for the phrase. See I. Sutherland. 1966. Computer graphics: Ten unsolved problems. *Datamation*, 12(5): 22–27.

ited time and time again as old problems were solved and new problems arose.[42] The article is arguably one of Sutherland's most famous works,[43] as it articulates a particular vision of the future for the field of computer graphics written by the then-director of the IPTO, a set of key problems that suggest a trajectory that researchers might follow.

The article is structured around what Sutherland then viewed as "a representative sample of the topics of interest to today's researchers in on-line computer graphics,"[44] and it consists of little more than a brief introduction followed by ten unnumbered sections, each detailing a single problem. Yet despite its relative brevity, the article would set the field's research agenda for over a decade. For Sutherland in 1966, the key unsolved problems in computer graphics were:

- Hardware Characteristics & Cost
- Problems of Technique
- Coupling Problems
- Describing Motion
- Halftone Capability
- Structure of Drawings
- Hidden Lines
- Program Instrumentation
- Logical Arrangement
- Working with Abstractions

42. See J. Blinn. 1998. Ten more unsolved problems in computer graphics. *IEEE Comput. Graph. Applic.*, 18(5): 86–89; M. Newell and J. Blinn. 1977. The progression of realism in computer generated images. *ACM '77 Proceedings of the 1977 annual conference*, pp. 444–448. ACM, New York; P. Heckbert. 1987. Ten unsolved problems in rendering. *Workshop on Rendering Algorithms and Systems*, Graphics Interface 87, Toronto, April 1987; M. Levoy. 1996. Open problems in computer graphics. Last updated April 8, 1966. Available at http://graphics.stanford.edu/courses/cs348b-96/open_problems.html; and J. Foley. 2000. Getting there: The ten top problems left. *IEEE Comput. Graph. Applic.*, 20(1).

43. Sutherland is best known for his *Sketchpad* work, but is also well known for his early writing on virtual reality and computer graphics. See I. Sutherland. The ultimate display. *Proceedings of IFIP Congress 1965*, pp. 506–508. Available at http://worrydream.com/refs/Sutherland%20-%20The%20Ultimate%20Display.pdf.

44. Sutherland, Ten unsolved problems, 22.

Examining these problems offers us a glimpse into the state of computer graphics at the very moment the field emerged as a coherent technical discipline, and the very same year that the Utah graphics program was founded. Perhaps surprisingly, the vast majority of the issues Sutherland outlined have little or nothing to do with image, picture, or vision. Instead Sutherland's concerns are for things like *Coupling Problems*, which dealt with the interfacing of graphical hardware with software and related systems; *Program Instrumentation,* which dealt with the standardization of program syntax and structure such that the relationship between parts of a program was clear and apparent; and *Hardware Characteristics and Cost*, which concerned the massive price tag of experimental hardware and the need to develop standardized tools that could be mass produced at a reasonable cost.

Only three of the problems deal specifically with the challenge of producing graphical object simulation itself: *Halftone Capability*, *Hidden Lines,* and the *Structure of Drawings*. *Halftone Capability* is the ability to shade solid objects so that they appear realistically opaque, which in turn leads to the simulation of surface effects such as lighting, color, and texture. *Structure of Drawings* deals with the standardization of object forms, and the kinds of geometric primitives that could be legible to a computer graphical program. By far the most visual concern is that of *Hidden Lines*, which dealt with a computer's ability to tell which lines in a given drawing should be made visible to a viewer.[45] These three problems outlined the minimal requirements for graphical realism in 1966, and they demanded a novel set of solutions not yet imagined.[46] Unsurprisingly, these problems map directly onto the three goals that Evans outlined in his own report on the state of his department in 1968, and from 1966–1970 these problems were the focus of a great deal of research at Utah, with multiple competing solutions developed for each.[47]

45. J. Gaboury. 2015. Hidden surface problems: On the digital image as material object. *J. Vis. Cult.*, 14(1): 40–60.

46. It is significant how distant these goals are from contemporary research into graphical realism, and how little explicit reference there is to the appearance of objects beyond the most basic attributes of visibility. As Jim Blinn noted in his keynote address at SIGGRAPH 1998, it wasn't until the 1970s that rendering became a primary interest for graphics research. At this stage researchers are concerned with "interaction problems," that is, the way objects and systems connect and interrelate. See J. Blinn. February 1999. "SIGGRAPH 1998 keynote address. *Conference Reports*, 32(1). Available at http://old.siggraph.org/publications/newsletter/v33n1/columns/conf.html.

47. One exemplary work is John Warnock's solution to the hidden surface problem, which was a major milestone of the time but was ultimately replaced by more efficient solutions such as Edwin Catmull's z-buffering. See J. E. Warnock. 1969. A hidden surface algorithm for computer generated halftone pictures. No. TR-4–15, University of Utah, School of Computing; E. Catmull.

While these problems may seem inconsequential when viewed through the lens of contemporary computer graphical challenges, it is important to note that many of the solutions that were first developed in the 1960s and 1970s continue to underpin a wide range of contemporary graphical techniques such as smooth shading, z-buffering, ray tracing, and more. The persistence of these solutions speaks directly to their historical significance, but it also helps make clear why narratives of technological development—what computer scientist and Utah graduate Jim Blinn (Ph.D. 1978) has called the "quest for realism"—have played such a powerful role in the evolution of the field toward more realistic and lifelike simulations. Through the trope of the "ten unsolved problems," realism becomes a problem that can be solved incrementally. As solutions to some problems are developed, a new set of ten problems would be published. Since Sutherland's initial article in 1966 an updated list has been published at least once a decade by various authors, suggesting that problem solving continues to be a primary framework by which the field is defined and research is framed. It begins in 1966 with ten unsolved problems; but to solve these problems Evans would need to recruit top-tier faculty and graduate students and begin to shape a community of researchers interested in making his vision of interactive graphics a reality.

11.5 Community

In its first five years the University would pull in dozens of researchers to this young program in Salt Lake City, although in 1968 the program came close to losing Evans altogether. After leaving the IPTO for Harvard in 1966, Ivan Sutherland remained in close contact with Evans and his family; and by 1967 the two began to discuss the possibility of founding a company to capitalize on the work being done at Utah.[48] Initially, they decided it would be best to move the venture East to the then-flourishing tech scene along Route 128 in west suburban Boston.[49] Despite the

1974. A subdivision algorithm for computer display of curved surfaces. No. UTEC-CSC-74–133, University of Utah, School of Computing.

48. David C. Evans Oral History Interview by Daniel Morrow. Computerworld Honors International Archives, p. 15.

49. There were also concerns over the potential conflict of interest with Evans' responsibilities at the University. Evans had worked specific language regarding his ability to perform contract work into his initial hire, but E&S served to magnify these concerns, leading to an explicit policy enacted by the University intended to marginally restrict commercial ventures such as E&S. See Proposed policy statement on conflict of interest. David C. Evans papers, Ms 625, Box 5, folder 1. Special Collections and Archives. University of Utah.

advantages Salt Lake City offered to a community of innovative researchers backed by the IPTO, it was far from the tech industry hubs of the time and the regional advantage they might afford.[50] Nonetheless, having just moved from California along with their seven children one year prior, the Evans family was not excited by the idea of leaving so soon. As Sutherland recalled:

> Joy Evans, Dave's wife, reasoned with him. She said, 'Dave, we have seven children, and Ivan has only two. It would be much easier for Ivan to move to Salt Lake than for us to move to Boston.' And I believe that was the seed of the University of Utah phenomenon.[51]

Sutherland moved with his family to Salt Lake City in 1968 to co-found the Evans & Sutherland Computer Corporation (E&S), accepting a tenured position at the University of Utah which he would hold until 1974. It is difficult to overestimate the centrality of E&S. It was the first computer graphics firm in the country, and many of the key technologies developed at Utah were produced either in partnership with E&S or transformed by E&S into commercial technologies. The company also served as an important resource for many of the Utah graduates who worked as employees for E&S both during their time as graduate students and/or on completing their degrees. E&S served as the platform from which a large number of Utah graduates launched their careers, and it is through the technologies produced by E&S that the research of the Utah program received wide circulation in the growing commercial graphics industry. The company was in effect a key partner organization with the Utah graphics program, and helped to both facilitate the transition from academia to industry in the 1970s, as well as to support a community of researchers beyond their careers as doctoral students at the university.

With Sutherland on board in 1968, Evans spent the next several years recruiting key faculty with the support of university president James Fletcher. Immediately following Sutherland Evans recruited Robert Barton, the chief architect of the Burroughs B5000 and inventor of the first stack machine architecture implemented in a mainframe computer. In 1969 he hired Thomas Stockham from MIT who began an innovative program of signal processing research that would lead to some of the earliest work in the field of digital recording. Stockham would later succeed Evans as department chair in 1972. In 1970 Evans recruited Charles Seitz from MIT, who

50. A. Saxenian. 1994. *Regional Advantage: Culture and Competition in Silicon Valley and Route 128*. Harvard University Press, Cambridge, MA.

51. H. K. Bowen and C. Purrington, April 28, 2006. The University of Utah and the computer graphics revolution. Harvard Business School, Unpublished Case Study, revised April 11, 2007), p. 5.

Figure 11.2 **Ivan Sutherland with David Evans at University of Utah.** (Courtesy of © University of Utah archives)

played a critical role in the development of asynchronous circuits and would later found the high performance interconnect and switching company Myricom. While several faculty would leave the University by the early 1970s, this rapid expansion was crucial to the program's success. In just a few short years Evans had amassed a distinguished faculty drawn from top-tier research institutions across the country. While the resources afforded by the university and the IPTO were a significant draw for faculty hires, a major key to this success was Evans' own leadership and enthusiasm for the work being done at Utah.[52] As John Warnock recalled,

52. Evans personal correspondence, along with interviews and oral history from faculty and graduate students, makes frequent mention of the natural resources of the Salt Lake City region, with particular interest in the accessibility of local ski resorts.

"Dave had the ability to get people excited. I just think he was a phenomenal motivator for getting people to come and work on important things."[53]

This penchant toward community building was also reflected in Evans' policies on graduate admissions, which were designed to seek out hard-working and creative students regardless of academic record. As Sutherland recalled,

> David was clever about admitting students who had unusual records. He decided that in every class we're going to admit a few outliers, folks who were clearly interesting, but for whom the academic system didn't exactly fit. That was quite a conscious admissions policy.[54]

Evans' goal was to create an environment in which students would thrive with few rules and as much responsibility as they could take on. In his first meeting with Evans in 1969, Al Davis (Ph.D. 1972)—now director of the University of Utah School of Computing—recalled asking what course requirements there were for the department. The answer: "There aren't any. Talk to the faculty, find out what you're interested in, then do great research and do great things."[55] This largely unstructured environment, along with the mentorship of Utah faculty, was key to the success of the department and its graduates.[56]

Students were allowed to form their own collaborations and were encouraged to focus narrowly on the key problems facing the field with the goal of finding broadly deployable solutions. These problems often served as the impetus for Ph.D. theses, many of which remain the *de facto* solutions for contemporary graphical software and hardware. The department's first student was Stephen Carr (Ph.D. 1969), who followed Evans from Berkeley and conducted early research on geometric mod-

53. Bowen and Purrington, University of Utah and the computer graphics revolution, 5.

54. Ibid., 8.

55. K. J. P. Lindberg. 2006. Pioneers on the digital frontier. *Continuum: The Magazine of the University of Utah*, Winter 2006–2007. Available at http://continuum.utah.edu/2006winter/feature3.html. This anecdote was also recounted in an interview with the author on March 21, 2012 in Salt Lake City.

56. It can be difficult to properly historicize a community of practice and the ways in which it contributed to a culture of innovation at a specific period in time. This is particularly true when that history is by necessity be colored by the nostalgia of living historical actors whose time at Utah was by many accounts fundamental to their later success. Oral history and interviews with Utah faculty and students are often filled with evocations of pioneering research and cowboy computing, claims whose narrative of exceptionalism mirrors the historical narrative of the Salt Lake City region itself. While it is important not to discount these narratives, it is equally important to remain critical of their historical utility.

eling.[57] Soon after in 1966 Evans recruited Alan Kay (Ph.D. 1969), whose graduate research would form the basis for his pioneering work on object-oriented programming and the graphical user interface design at Xerox PARC in the 1970s. Kay's atypical academic background speaks directly to Evans' admissions policies during this period. In 1961 Kay had been expelled from Bethany College in West Virginia after participating in a protest of the university Jewish quota before joining the volunteer air force to work as a computer programmer, returning to finish his B.S. in Mathematics and Molecular Biology at the University of Colorado Boulder.

Ivan Sutherland also played a significant role in graduate student recruiting during this period. While Evans was setting up the department in Salt Lake City, Sutherland made multiple trips to MIT to recruit potential graduates. He also travelled internationally to France, where he believed the most innovative mathematical research was being done,[58] and recruited several key graduates from the National Institute for Research in Computer Science and Control (INRIA) in Rocquencourt and the École Polytechnique in Palaiseau.[59] Most notable among them were Henri Gouraud (Ph.D. 1971) and Bui Tuong Phong (Ph.D. 1973), each of whom developed unique solutions to the problem of object shading still widely in use today.[60]

The department also drew in local students from the Salt Lake City area, many of whom had not initially intended to pursue computing or graphics, but found themselves at the center of Evans and Sutherland's research community. Early Utah locals included Nolan Bushnell (B.S., 1969), who would later found Atari, Inc. in 1972, and Alan Ashton (Ph.D. 1970), who developed WordPerfect in 1979. Evans and Sutherland frequently poached promising students from other departments, such as physics or mathematics, to join the growing computer science program. John Warnock (Ph.D. 1969) recalls that, "[after] working for about three years on a Ph.D. in mathematics . . . Dave took all of the mathematics and applied it

57. For a comprehensive list of Utah graduates, see Ph.D. alumni, School of Computing, University of Utah. Available at http://www.cs.utah.edu/~sgoyal/soc/phd.html (accessed January 2016).

58. During the 1970s and 1980s French computer scientists would make significant contributions to the field of computer graphics. Among them are Gourard and Phong, but the list would also include Pierre Bézier—one of the founders of the fields of solid, geometric, and physical modeling as well as in the field of representing curves, particularly in CAD/CAM systems—and Benoit Mandelbrot, who popularized fractal geometry.

59. Jim Crow, Unpublished email interview with Peter Shirley, November 8, 1996.

60. H. Gouraud. 1971. Computer display of curved surfaces. No. UTEC-CSc-71–113, University of Utah, School of Computing; B. Tuong-Phong. 1973. Illumination for computer-generated images. University of Utah, School of Computing.

toward the coursework for the College of Engineering's Computer Science Ph.D. Consequently, I've never had a course in engineering."[61] Warnock's dissertation produced a key solution to the hidden surface problem described above. He would go on to work at E&S and Xerox PARC in the 1970s before founding Adobe Systems in 1982.

In these first five years the department thrived. With a steady influx of funding and researchers they were able develop workable solutions to a large number of key problems with the full support of the IPTO, whose cavalier attitude over funding and resources facilitated rapid expansion and innovation. As Alan Kay recalled in one particularly colorful anecdote:

> I remember once, after being at Utah for three or four months, Dave was saying in the midst of some other conversation that was going on, "We're almost out of money. Got to go get some more." So he got on a plane, and three days later came back with our next ARPA contract. That was the way they did it in those days. People did not generally write big proposals.[62]

But by the early 1970s all this began to change, due in no small part to the passing of the Mansfield Amendments by Congress in 1970 and 1973. The two amendments, named after Montana Democratic senator Mike Mansfield, severely limited the use of military funds for the kinds of long-term visionary grants that were the hallmark of the Licklider-era IPTO. If a project had no explicit military application, it could no longer be funded by the Defense Department. This newfound administrative oversight crippled many of the Centers of Excellence that had been founded over the previous decade, and had a dramatic impact on the future of the Utah program. Applications and reports to the Defense Department grew noticeably in length and complexity during this period, as funding became much more difficult to acquire. The old boys club of the 1960s was beginning to come under heavy scrutiny, despite the demonstrable success of many IPTO initiatives. This transformation is clearly visible in an exemplary letter from the program manager for the DARPA Cybernetics Technology Division to David Evans in 1977, in which he chides Evans for his proposal on software and database production:

61. Bowen and Purrington, University of Utah and the computer graphics revolution, 9.

62. Ibid., 6.

> I appreciate short, no baloney proposals, but this one may have gone a little too far. I need some more information. (You might also want to clean up the spelling mistakes and use the correct name of our agency.)[63]

What follows is a list of 12 bullet points to be answered, and a request for a revised budget sheet.

Of course, research into computer graphics did not stop. It simply changed venues. As ARPA funding dried up, the IPTO's Centers of Excellence began to hemorrhage researchers, who in many cases turned to jobs in the burgeoning computer industry. One notable exception was the graphics group at the New York Institute for Technology, a largely self-funded initiative driven by the eccentric vision of its founder Alexander Schure. At NYIT Ed Catmull (Utah Ph.D. 1974) and Alvy Ray Smith made early strides in computer animation prior to their move to Industrial Light and Magic in 1979 and the founding of Pixar in 1986.[64] With the founding of ACM SIGGRAPH in 1969 and the first annual conference on computer graphics in 1974 the field began to professionalize and move beyond the largely academic context of its first decade. Over the course of the 1970s SIGGRAPH and the ACM served as key venues for graphical research, and helped support a growing network of researchers transitioning from early unsolved problems to commercially deployable applications. In this sense the ACM served as a bridge between research and industry at the very moment in which the field was transitioning away from academia toward new funding and research models.

Over the course of the 1970s research centers at Xerox PARC, NASA's Jet Propulsion Laboratory, and the New York Institute for Technology were populated with Utah faculty and graduates who brought with them years of experience, as well as research results that could be capitalized on by corporate initiatives. Evans & Sutherland continued to grow as well, due in part to its ability to secure Defense Department funds for research and development in the emerging field of military and commercial flight simulation. By the end of the decade David Evans had largely left academia to pursue commercial graphics full time, resigning as chair of the

63. This final jab refers to the 1972 name change from ARPA to DARPA, adding the word "Defense" to signal this shift in priorities. See Letter to David Evans from Craig Fields, David C. Evans papers, Ms 625, Box 89, folder 2. Special Collections and Archives. University of Utah.

64. See A. R. Smith. 2001. Digital paint systems: An anecdotal and historical overview. *IEEE Ann. Hist. Comput.*, 23(2): 4–30.

Computer Science Division of the University of Utah in 1972 before leaving the university entirely in the early 1980s.[65] By 1979 the golden era of the Utah program was over, but the network of researchers that began their careers at Utah was firmly established, and would grow to have a massive impact on the field of computing as it became increasingly graphical and interactive.

11.6 Other Places[66]

The history of the Utah graphics program is significant for many reasons. First, it offers an exemplary history of institutional success in which funding, leadership, administration, and talent all aligned to jump-start a field of inquiry decades ahead of its time. I have attempted here to articulate precisely why this happened and how each of these key players contributed to the success of computer graphics at Utah during this brief but influential period. Nonetheless the Utah story points to something much larger than this 15-year regional history. Looking back it may be surprising to some that a program so geographically isolated from the technical research and economic centers of the U.S. could succeed in the way that it did, but as this chapter has shown the program was by no means outside the critical networks of influence that funded and fed computer science during this early period.

Nonetheless, the field of computer graphics as a whole is dominated by these secondary sites, by spaces where we might not expect technological development to thrive. From some of the earliest experiments in vector graphics by artist and draftsman Ben Laposky of Cherokee, Iowa, in 1950,[67] to the first use of the term “computer graphics” in 1960 by Verne Hudson and William Fetter while working for Boeing in Wichita, Kansas,[68] the field is dominated by individuals and groups that

65. Resignation Approval Letter to David Evans, December 12, 1972, Box 18, Folder 12, Coll. 0199, James C. Fletcher Presidential Records, 1937–1971, University of Utah.

66. This language of “othering” is here used primarily within the context of the history of computing, although I acknowledge its significance to a much broader set of discourses, particularly in critical race studies and gender studies, where authors such as Edward Said (1978) and Simone de Beauvoir (1949)—among many others—have discussed the myriad ways in which populations and individuals are politically, socially, culturally, and economically othered in their difference. This same discourse could be applied to the many ways in which Utah marks an “othered” space in the history of technology, namely its rural setting, its foundational fantasies of manifest destiny, and the strong presence of Mormonism as both a religion and a culture during this period.

67. B. F. Laposky. 1969. Oscillons: Electronic abstractions. *Leonardo*, 2(4): 348.

68. W. Fetter. 1982. A progression of human figures simulated by computer graphics. *IEEE Comput. Graph. Applic.*, 2(9): 10.

fall outside of the narratives of research and innovation that traditionally dominate the history of computing. Likewise, the field of computer graphics itself quickly escapes any single narrative of development as it grows and evolves to suit the needs of a wide range of disciplines and interests. Fields as diverse as computer animation, digital gaming, graphical interfaces, computer-aided-design, 3D printing, word processing, desktop publishing, data visualization, graphic design, architecture, biology, and medicine have been shaped by computer graphics over the past 50 years, and have adapted graphics in various ways to serve their own unique needs for man-machine communication. Over the course of its history, computer graphics has grown to become a medium much larger than those few key problems that Evans and others set out to solve in 1966, and as such it is crucial that we continue to attend to its history.

12

Framing Computer Security and Privacy, 1967–1992

Rebecca Slayton

Computer security is multivalent. It has, at various times, been seen as crucial to protecting individual privacy, preventing criminal activities, maintaining the safety of critical infrastructure, and preserving national security. Proposed methods for achieving security range from the highly technical to the legal and managerial. Similarly, responsibilities for computer security have been allocated in diverse ways, sometimes focusing on the computer researchers, the computer industry, system administrators, and policymakers.

This chapter analyzes how ACM leaders, special interest groups, conferences, and publications framed computer security from the mid-1960s through the early 1990s, focusing on how they answered three kinds of questions. What were the *risks* of insecure computers—that is, in what ways were they vulnerable, who threatened them, and what were the dangers of insecurity?[1] What *methods* could best secure computers—better technical designs, management, legislation? And how should *responsibility* for security and privacy be allocated—what roles did individual computer professionals, the computer industry, and policymakers have to play? Security framings were not necessarily cohesive, nor did they always give equal or systematic treatment to these elements. Some were organized around technological solutions, while others were organized around particular risks or responsibilities. However,

1. Although cybersecurity risks today are frequently treated as the product of threats, vulnerabilities, and consequences, historical actors used such terms interchangeably.

security framings tended to direct attention to particular risks, solutions, and responsibilities, and not others.

By examining how ACM leaders and members have framed computer security, and how those framings have changed over time, this chapter aims to broaden understanding of what computer security has historically meant, while helping to situate existing historical work on computer security. We have a number of studies that focus on a particular way of framing computer security, but very few have examined the relationships between and among them. For example, Donald MacKenzie examined the development of a particular set of methods—formal specification and verification—as a way of making computers more reliable and secure.[2] Steven Levy detailed the development of a very different method—cryptography—as a method of protecting privacy.[3] Other historians have focused on the development of government standards or policies on computer security.[4] However, we know relatively little about the relationships between and among these various approaches to security.

ACM has left a rich historical record for examining these interrelationships. ACM journals and the newsletters of its special interest groups (SIGs) provided forums for discussing both technical and political aspects of computer security. ACM helped to develop educational standards which shaped new generations of computer professionals and sometimes contributed to research. ACM conferences and workshops also helped to connect professionals in business, law, and other fields with computer scientists. At times, ACM worked directly with government agencies such as the National Bureau of Standards (NBS). ACM was not the only professional society to offer such resources; the IEEE Computer Society played an

2. D. MacKenzie. 2001. *Mechanizing Proof: Computing, Risk, and Trust*. MIT Press, Cambridge, MA.

3. S. Levy. 2001. *Crypto: How the Code Rebels Beat the Government—Saving Privacy in the Digital Age*. Penguin Books, New York.

4. J. Yost. 2007. A history of computer security standards. In K. De Leeuw and J. A. Bergstra, editors. *The History of Information Security: A Comprehensive Handbook*, pp. 595–621. Elsevier, Amsterdam. See also essays in a special issue of *IEEE Annals of the History of Computing:* J. R. Yost. 2015. The origin and early history of the computer security software products industry. *IEEE Ann. Hist. Comput.*, 37(2): 46–58; R. Slayton. 2015. Measuring risk: Computer security metrics, automation, and learning. *IEEE Ann. Hist. Comput.*, 37(2): 32–45; M. Warner. 2015. Notes on the evolution of computer security policy in the US government, 1965–2003. *IEEE Ann. Hist. Comput.*, 37(2): 8–18; S. B. Lipner. 2015. The birth and death of the Orange Book. *IEEE Ann. Hist. Comput.*, 37(2): 19–31; D. Park. 2015. Social life of PKI: Sociotechnical development of Korean public-key infrastructure. *IEEE Ann. Hist. Comput.*, 37(2): 59–71; L. DeNardis. 2015. The Internet design tension between surveillance and security. *IEEE Ann. Hist. Comput.*, 37(2): 72–83.

important role, as did the American Federation of Information Processing Societies (AFIPS), the umbrella organization of which ACM was part, and which sponsored the Spring and Fall Joint Computer Conferences, and then the National Computer Conference.

This chapter draws on archival research, particularly the papers of Paul Armer at the National Museum of American History (hereafter referred to as the Armer collection). It also uses original analyses of ACM publications. Keyword searches of the ACM Digital Library were used to gather data on ACM publications related to "security" or "privacy" appearing between 1967, when security became a major topic of discussion by computer scientists, and 1980, around the time that groups began specializing in computer security.[5] Although the resulting list was not comprehensive, it provides some insights on ACM's activities in this area. As Figure 12.1 shows, computer security and privacy were mainstream issues for ACM; the largest fraction of papers appeared in ACM's mainstream publications or conferences, namely the *Communications of the ACM*, *Journal of the ACM*, the ACM Annual meeting, or AFIPS conferences which ACM co-sponsored. However, substantial fractions of papers also appeared in publications or conferences sponsored by ACM's special interest groups (SIGs)—particularly SIGOPS.

To put these papers in context, I surveyed all issues of three ACM Special Interest Groups—Operating Systems (SIGOPS), Computers and Society (SIGCAS), and Security, Audit and Control (SIGSAC)—from their founding through 1992, identifying the primary themes and issues that organized their discourse on computer security. These surveys identified additional papers not discovered through the keyword search of the Digital Library but nonetheless relevant to security. Additionally, I surveyed all issues of *Commununications of the ACM* from 1967–1991 for discussion of computer security. To better understand the relationship between ACM and public policies on computer security, I also surveyed all U.S. Congressional hearings on computer security or privacy from 1965–1995, with focused attention on the statements of ACM leaders. I supplemented this document-based research with oral histories and interviews.

5. Papers were initially gathered using the keyword search "secure" or "security," but were further screened to eliminate results that did not explicitly discuss computer security or related concepts (such as privacy or protection) in the text. Papers that only mention security in the author affiliation (e.g., "National Security Agency"), in references, or in other contexts (such as financial "securities") are not included in the data shown here. Additionally, papers which only mentioned security in passing were not included.

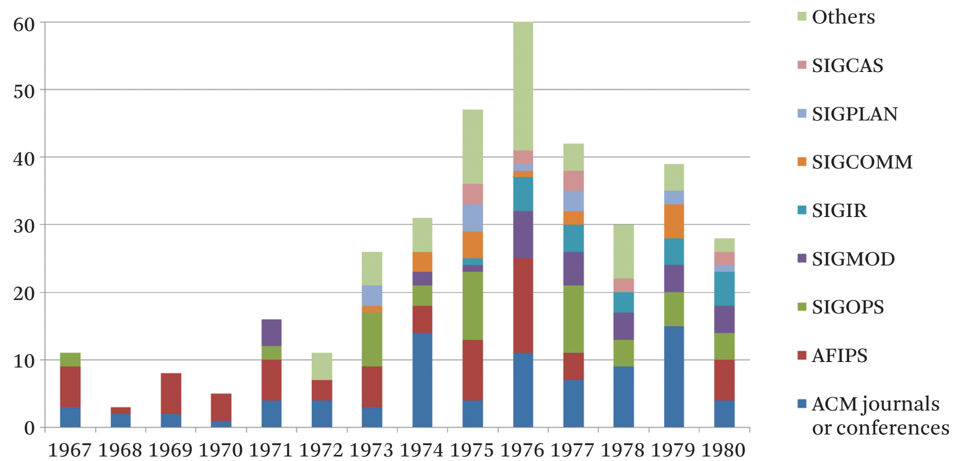

Figure 12.1 **Articles in ACM publications on security or privacy, 1967–1980.**

I argue that ACM leaders and special interest groups (SIGs) developed distinctive ways of framing computer security and privacy in the 1970s, but began to dispute those framings in the 1980s as computer networking became more common and as ACM became more involved with public debate about computer security and privacy. Although changing technology prompted debate about the appropriate way to frame computer security, technology by no means determined the ways that computer security was envisioned. Rather, ACM leaders and members maintained a variety of conceptions of risks, solutions, and responsibilities.

The remainder of this chapter is organized into three parts. The first half discusses how ACM leaders and special interest groups framed computer security during the 1960s and 1970s, during the development of time-shared computing. By enabling multiple users to share computer resources nearly simultaneously, time-shared computing could dramatically reduce costs, but it also came with new security and privacy risks. ACM's SIGs encouraged distinctive framings of these risks, as each SIG tended to approach security or privacy as a part of a broader research agenda. SIGCAS tended to focus on the threats that large bureaucratic organizations, whether government or private sector, posed to the freedoms of individual citizens and consumers. By contrast, SIGOPS and related groups focused more on risks to the organizations developing and using time-shared computers, whether they were in the private sector, or government or military. While SIGCAS focused on legislative or ethical solutions to the problem of individual privacy, SIGOPS and

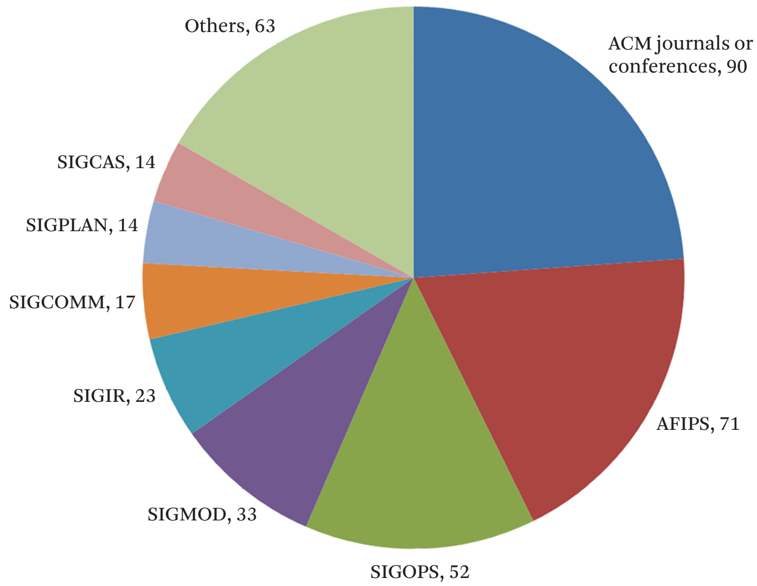

Figure 12.2 **Articles in ACM publications on security or privacy, 1967–1980.**

related groups sought technological solutions to the organizational problems of security. While these distinctive ways of framing security were not necessarily in conflict, SIGs studied solutions to these problems largely independently, as a part of distinctive research agendas.

In a brief second section, I discuss two major changes that took place in the early 1980s: new groups specializing in computer security, most notably the IEEE Symposium on Security and Privacy, helped to overcome some of the silos inadvertently created by ACM SIGs. And second, increased computer networking raised questions about the appropriate methods for achieving security.

The third part of this paper discusses how the increased use of computer networks made computer security a more salient public issue, even as it sparked debate about the appropriate way to frame computer security and privacy. As journalists began to focus more attention on computer security and policymakers proposed laws that could significantly affect the activities of computer professionals, ACM became more involved in public debates about security, and ACM members began to debate the appropriate way to frame computer security. ACM leaders eventually institutionalized the Computers, Freedom, and Privacy conference as a forum for discussing such questions.

12.1 Framing Record-Keeping Security

12.1.1 The Relationship between Privacy and Security

Perhaps no ACM leader influenced the framing of computer security more than Willis Ware, the head of the Computer Science Department at RAND. Ware led one of the first public discussions of computer security during two sessions of the Spring Joint Computer Conference (SJCC) in 1967. Historians have emphasized the panel's national security context; the discussion was remarkable for its explicit discussion of the National Security Agency (NSA) at a time when the agency's identity was not public; it was more likely to be referred to as "No Such Agency."[6] Ware himself was also deeply involved in studies of military computer security at RAND, and chaired a widely-cited Defense Science Board study that was completed in 1970, "Security Controls for Computer Systems" (better known as the Ware report).

But while military security was certainly a consideration of the 1967 AFIPS sessions, they were not solely, or even primarily, focused on the problems of military security. One panel member represented the commercial time-sharing firm Allen-Babcock Computing, and the much more visible context was a highly public debate about the U.S. Bureau of the Budget's 1966 proposal for a National Data Center and the corresponding risks to privacy. A *Commununications of the ACM* staff reporter framed Ware's panel in the context of Congressional hearings about the data center, which were chaired by Representative Cornelius Gallagher (New Jersey), who delivered the SJCC keynote address.[7] Similarly, Alan Westin, a law professor at Columbia University who was leading a study of privacy issues raised by a wide range of new technologies, discussed the need for legislation to protect privacy. Thus, in the late 1960s, the focus of public discourse about computer-related insecurity was not on threats to national security or even crime, but on the Orwellian specter of government surveillance.

In 1967, Ware distinguished these different contexts by explaining that for the purposes of discussion, "'security' and 'classified' will refer to military or defense information or situations; 'private' or 'privacy,' to the corresponding industrial, or non-military governmental situations."[8] However, Ware's conception of security

6. See, for example, MacKenzie, *Mechanizing Proof*, 16, 155.

7. J. Titus. 1967. Security and privacy. *Commun. ACM*, 10(6): 379–380.

8. W. H. Ware. 1967. Security and privacy in computer systems. In *Proc. of the April 18–20, 1967, Spring Joint Computer Conference AFIPS '67*, pp. 279–280. ACM Press, New York.

and privacy began to change as he took on studies of the privacy problem in civilian government agencies. In 1972 he became the chair of an advisory committee on automated data systems for the U.S. Department of Health, Education, and Welfare (HEW), which delivered its report in July 1973.[9] Perhaps as a result of this study, or his role as chair of a presidentially appointed Privacy Protection Study Commission, which delivered its final report in 1977, Ware's conception of security and privacy shifted in the mid-1970s.[10] Rather than arguing that security and privacy dealt with different *contexts*, he began to argue that security dealt with *techniques*, and privacy with *social expectations*.

For example, in testimony on electronic funds transfer systems, Ware defined security as "the totality of safeguards required . . . to protect computer based systems" against "deliberate or accidental damage," "denial of use" by their rightful owners, or "divulgence to or use [of data] by unauthorized persons." By contrast, Ware viewed privacy as "a troublesome term because it is broad in scope." He suggested that privacy is the "social expectation" of three things: "that an individual must be able to determine to what extent information about himself is communicated to or used by others"; "that an individual will be protected against harm that might occur because of the information held about him in some record system," and "that the individual will be protected against unwelcome or intrusive [data] collection."[11]

In Ware's conception, security and privacy were both social and technical issues. An article in the SIGCAS newsletter described how Ware's committee realized this while studying the privacy problem for HEW:

> Although most members of the committee began the study with the belief that technology was already available or could easily be developed to protect the privacy of individual information in computer systems, continuing evaluation brought the group to the conclusion that technology and technologists would not solve the problems or overcome the harmful effects of such systems.[12]

9. Available at www.justice.gov/opcl/docs/rec-com-rights.pdf.

10. Final report is available at epic.org/privacy/ppsc1977report/c1.htm. The committee aimed to remedy several weaknesses in the way that the Privacy Act of 1974 had been implemented, and it articulated a somewhat broader and more refined set of principles for fair information practices.

11. W. H. Ware. 1977. Testimony before the National Commission of Electronic Fund Transfer. *Comput. Soc.*, 8(1): 2–3.

12. C. Landis. 1973. Computer records and people. *Comput. Soc.*, 4(3): 15.

Figure 12.3 **Security pioneer Willis Ware at National Computer Conference.** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

The reason was that most managers of record-keeping systems "did not appreciate the threat to personal privacy generated by the systems," and even those that did recognize this threat did not understand "the type or extent of protection offered by the system." Furthermore, "the policymaker expected the computer technician to take responsibility for" protecting privacy, while most computer technicians "denied any responsibility for providing protection of individual data records," explaining that they were "only following orders from higher level managers or policymakers."[13]

13. Ibid.

The HEW study ultimately proposed a Code of Fair Information Practices which became quite influential.[14] It had a major influence on the Privacy Act of 1974, which required that federal agencies publicize the existence of databases holding personal information; provide a means for individuals to learn about and correct their records; ensure that personal data not be shared without consent, and that data not be corrupted or misused.[15] The Privacy Act also called for the creation of the Privacy Protection Study Commission to investigate privacy issues in the private sector, and Ware helped to lead this commission.

By the late 1970s, Ware emphasized that responsibility for privacy lay with the organizations charged with managing such systems. In Congressional testimony, he explained:

> A recordkeeping organization cannot say to the vendor that supplies its computer hardware and software: "Solve my privacy problem." It is an unreasonable request and a demand impossible to respond to. Whatever responsibilities will be imposed by law upon keepers of information about people must fall upon the recordkeeper and cannot be passed off to the vendors who happen to supply equipment to recordkeepers.

On the other hand, Ware also argued that the government should provide computer vendors with incentives to improve technical security features: "As soon as it becomes clear what legislative requirements must be met by recordkeepers, the pressure of their market demands will be sufficient to cause the computing industry to respond adequately."[16]

In short, while privacy tended to be framed as a "social" issue, and security as a "technical" issue, Ware recognized the interaction between social and technical factors. Government policies could have a very direct impact on technological developments, which could in turn shape the ability of computer managers to implement appropriate solutions to the privacy problem. For ACM, this raised the question of how, exactly, the profession should be involved in public debate over an issue that was not narrowly technical.

14. Secretary's Advisory Committee on Automated Personal Data Systems. 1973. Records, computers, and the rights of citizens. U.S. Department of Health, Education, and Welfare: Education U.S. Department of Health, and Welfare.

15. For the 1974 Privacy Act, see www.justice.gov/opcl/overview-privacy-act-1974–2012-edition.

16. W. H. Ware. 1977. Impact of telecommunications technology on the right to privacy. *Comput. Soc.*, 8(4): 5.

12.1.2 Privacy and the Establishment of SIGCAS

Paul Armer, Ware's colleague at RAND, was deeply involved in questions of how ACM should tackle social dimensions of computing. Armer served as an ACM Council member from 1964–1968 and as chair of an ACM Special Interest Committee (SIC) on social responsibilities in the mid-1960s, in addition to occupying leadership positions in AFIPS. Armer, like Ware, participated in Congressional debates over privacy policy. In 1968 he testified "that privacy lacks an organized constituency," noting that Westin's work on privacy was funded by the Carnegie Corporation, while the "little work that has been done at Rand has either resulted from related work on military security or has been supported by Rand Corp . . . " Armer argued that the "forces of the marketplace are apt to have little impact in the near future on improving the state of privacy in our society," and that the market "needs prodding from the Government."[17]

Although he did not testify as a representative of ACM, Armer thought very carefully about how ACM should be involved in shaping government policies. He issued a strong response to the Washington, DC chapter of the ACM in 1967, when it passed a resolution opposing the national data center and forwarded it to the ACM national membership for consideration. The resolution questioned "whether a possible loss of individual rights, however small, is justified in providing a probable gain, however large, in government efficiency and economy," and concluded that "in a democratic society individual rights take precedence and determine a practical upper bound on the efficiency obtainable in government."[18] Armer was among several ACM leaders who were very critical of the resolution. He was "dead set against" it because posed a false dichotomy "between government efficiency and the individual's privacy." He noted that as "Alan Westin has frequently pointed out, the problem at hand is to achieve the appropriate balance between the individual right to privacy and society's right of discovery."[19]

Armer was not the only ACM leader to object to the resolution. Donn Parker, a computer scientist at SRI who held leadership positions in ACM during the 1960s and 1970s, objected that "ACM must limit itself to giving advice and taking posi-

17. Senate Judiciary Committee. February 6, 1968. *Computer Privacy*, p. 331. Ninetieth Congress, 2nd session.

18. Washington DC Chapter resolution. May 4, 1967, Box 2, Folder "ACM Council Meeting Binder, Monday, August 28, 1967." Paul Armer Collection, National Museum of American History (hereafter Armer Collection).

19. Armer to Pete Warburton. July 5, 1967. Box 2, Folder "ACM Council Meeting Binder, Monday, August 28, 1967," Armer Collection.

tions on issues only within the areas of its competence," and that the resolution took "liberties" on non-technical issues. For example, he felt that ACM was competent to discuss "the possible technical techniques of unauthorized penetration of a private data bank and the amount of work required to succeed," but not "the consequences of unauthorized penetration."[20] He warned that the resolution "could make a precedent resulting in such extremes as taking sides in the Viet Nam war issues. . . ."[21]

As this suggests, beneath the surface of this debate were anxieties about the politicization of ACM during dispute over the Vietnam War.[22] As R. Arvid Nelsen and Janet Toland note in their respective chapters in this volume, members of ACM did not always agree upon the ACM's scope of engagement with political or social issues. In 1969, after the ACM Council voted to dissolve its Special Interest Committee on Social Implications of Computing (SIC)^2 because of inactivity, many computer professionals protested what they viewed as an effort of the ACM "establishment" to limit ACM activities on controversial social issues such as the Vietnam War.[23]

By fall 1969, ACM responded to such concerns by announcing plans for a new SIG on Computers and Society (SIGCAS). The group began by establishing three subcommittees, respectively on "data banks and privacy," "jobs and automation," and "education of noncomputer people about computers."[24] As Janet Toland notes in her chapter, members of SIGCAS had diverse interests and views on their group's mission. Nonetheless, privacy provided a focal issue for SIGCAS at a time when the committee did not agree on its mission. In 1970, the group discussed "what

20. Parker to Oettinger. June 30, 1967. Box 2, Folder "ACM Council Meeting Binder, Monday, August 28, 1967," Armer Collection.

21. See also George Sadowsky to Tony Oettinger. June 27, 1967. Box 2, Folder "ACM Council Meeting Binder, Monday, August 28, 1967," Armer Collection.

22. Issues came to a head in the late 1960s after several members of the ACM called for the association to issue a statement against the Vietnam War. Instead, ACM referred a "question of importance" to the general membership: "should the constitution of the ACM be revised to permit comment on deeply social or political issues?" The question was soundly defeated. B. A. Galler. 1969. ACM President's letter: I protest. *Commun. ACM*, 12(8).

23. See, e.g., the Computer Professionals for Peace (CPP) Newsletter. *Interrupt*, May 1969. D. McCracken papers, CBI 43, Charles Babbage Institute.

24. The Computer Professionals for Peace noted these three committees in its newsletter, *Interrupt*, in November 1969. Dan McCracken papers, CBI. To CPP, this signaled that ACM wanted a "'safe' committee which will avoid controversy . . . " These three subcommittees had also filed interim reports by April of 1970; see Subcommittee reports. *Comput. Soc.*, 1(2).

SICCAS is and should be," but without reaching a conclusion. The SIGCAS chair reported: "Some feel we should be an action group; some a discussion group; some an education group."[25] Two years later, SIGCAS members still "expressed concern about the lack of an articulated intellectual basis."[26]

However, nobody disputed that privacy was a central issue for the group. The privacy debate gave the group new visibility in Washington. The first chair of SIGCAS, Robert P. Bigelow, was a lawyer who testified in Senate hearings on federal databanks in 1971.[27] One of the earliest SIGCAS newsletters featured a "broadside" by Congressman Gallagher, who exhorted SIGCAS members to take responsibility for protecting privacy:

> If we could characterize Eichmann as an inhuman monster because he got full trains to Buechenwald on time, perhaps we should take a closer look at computer experts who, seduced by efficiency and economy, into doing the cheapest and quickest job, take no professional action to deny unauthorized access in systems containing all information about everybody.[28]

Over the next decade, the SIGCAS newsletter took on many issues with privacy and security implications, including electronic voting machines, electronic funds transfers, courses on the social implications of computing, and safety-critical computing. Its most consistent focus was on the threat that large bureaucratized databases posed to the individual privacy and freedoms of citizens and consumers, and legislative solutions to the problem.

In the late 1970s and early 1980s, SIGCAS also began to address the issues of computer-related crime—which generally assumed that individual criminals posed a threat to government or private sector organizations. This perspective was very clear in perhaps the most prolific and influential researcher on computer crime, Donn Parker. In a 1979 article, Parker argued that new applications of computers, such as electronic funds transfer systems and automatic teller machines, were changing "the nature, needs, and methods of trust," with the effect that "the nature and methods of committing fraud" were also changing.[29] He argued that it was an open question as to whether fraud would become easier or more difficult,

25. R. P. Bigelow. 1970. From the chairman. *Comput. Soc.*, 1(2): 2.

26. ACM 1972 SIGCAS business meeting. *Comput. Soc.*, 3(3): 8.

27. Senate Judiciary Committee. *Federal Data Banks, Computers, and the Bill of Rights: Part I*. Ninety-second Congress, First Session, Feb. 23–25, Mar. 2–4, 9–11, 15, 17, 1971.

28. C. Gallagher. 1970. The computer as Rosemary's baby. *Comput. Soc.*, 1(2): 4.

29. D. Parker. 1981. The impact on computers on trust in business. *Comput. Soc.*, 11(3): 42.

but that fraud could be reduced “by eliminating system weaknesses, increasing controls that prevent or detect fraud-related events, and decreasing the number of people who have the skills, knowledge, and access to make system changes without detection.”[30] Parker, who had been active in ACM committees on professional standards and ethics, also raised the question: “How will changes in the need for trust change business transactions and business ethics?”

The focus on ethical solutions to the problems of computer crime was characteristic of SIGCAS, and encouraged the dominant conception of computer crime as a threat that individuals posed to businesses. This perspective was encouraged by Jay “Buck” Bloombecker, a lawyer who established the National Center for the Analysis of Computer Crime Data in the early 1980s, around the same time that he was elected to be Secretary-Treasurer of SIGCAS. Bloombecker frequently published articles in the SIGCAS newsletter suggesting that computer crime was an ethical and legal problem.[31] For example, in a 1983 SIGCAS article, he cargued that “the role of law . . . is to put conscience in computing.”[32] This emphasis on individual ethics encouraged the view that computer-related crime was a threat individuals posed to businesses.

However, some members of SIGCAS challenged this framing. In 1981, Rob Kling highlighted some weaknesses in a growing literature on computer crime:

> First, most of the cases examined are those in which businesses are victims; the perpetrators are individuals or small groups acting in relative isolation and pursuing idiosyncratic criminal ventures. Cases in which computer systems are instruments of businesses acting against their clients, e.g., consumer fraud, are largely ignored. Second, these events are typically removed from the social worlds in which they occur and simply labelled as “abuses” or “crimes,” e.g., invasions of privacy, fraud.[33]

30. Ibid.

31. For example, in winter 1982, *Computers & Society* 12(1) featured three of Bloombecker’s articles: Computer crime—career of the future?, Employee computer abuse—what to do?, and Who are the computer criminals?

32. B. Bloombecker. 1983. Conscience in computing, a Law Day perspective on computer crime. *Comput. Soc.*, 13(3): 11. See also B. Bloombecker. 1987. Computer ethics: An antidote to despair. *Comput. Soc.*, 16(4): 3–11; B. Bloombecker. 1990. Needed: Binary Bar Mitzvah’s and computer confirmations. *Comput. Soc.*, 20(4): 20–21.

33. R. Kling. 1981. Computer abuse and computer crime as organizational activities. *Comput. Soc.*, 11(4): 13.

Kling's article underscored the potential for corporate malfeasance in routine organizational activities—for example, by making billing errors in the company's favor, and making those errors difficult to correct. Kling also pointed to ambiguities around the "crimes" perpetrated by individuals: "the use of 'spare' computer time for private, recreational purposes by a computer programmer may be viewed as a theft of private property, or merely as a job perquisite akin to using a company telephone for limited personal calls." He noted different opinions "over whether a particular act should be labelled as an 'abuse,' and furthermore, even if labelled an abuse, whether it should be prohibited by law."[34]

In summary, although SIGCAS featured multiple perspectives in the 1970s and early 1980s, it focused primarily on the threat that organizations posed to individual freedoms and liberties, and on legislative solutions to the problem. By contrast, as the next section shows, ACM's Special Interest Group on Operating Systems (SIGOPS) focused more on threats to the organizational and corporate interests, and technical solutions to those threats.

12.1.3 SIGOPS and the Concept of Protection

SIGOPS was established in the mid-1960s as a group interested in time-shared computing.[35] Members of SIGOPS recognized that security was crucial to creating a new industry around the information utility from the very beginning. At the group's first Symposium on Operating Systems Principles (SOSP) in 1967, Jack Dennis, who was working on the Multics time-sharing project at MIT, began his paper by noting that "the effective operation of free enterprise in creating the envisioned information service industry is dependent on three accomplishments," one of which was "the development of public, message-switched communications services with adequate provisions for information security."[36]

Dennis's colleague at MIT, Robert Graham, painted a similar vision. Anticipating that the information utility would "probably include users who are competitive commercially," he explained:

> The system will be used for many applications where sensitive data such as company payroll records will need to be stored in the system. On the other hand, there will be users in the community who wish to share with each other data and procedures. . . . Service bureaus—software producing companies—and other service

34. Ibid.

35. For major events in SIGOPS history, see www.sigops.org/history.html.

36. J. B. Dennis. 1967. A position paper on computing and communications. In *Proc. of the First ACM Symposium on Operating System Principles*, 6.1.

> organizations will have procedures which they wish to rent. Some groups may rent access to data bases. Finally, there will be public libraries of procedures supplied by the information processing utility management.[37]

As these words suggest, Dennis and Graham were among SIGOPS members who focused on a vision of commercial time-sharing, rather than military computing. In fact, aside from a description of the "military security principle of 'need to know'," Graham's discussion made little mention of either computer security or the military.[38]

Instead, he focused on the challenge of "protection," or the problem of controlling access to the operating system. The problem of "access control" or "protection" was aimed at more than just malicious users; it was also seen as something that could increase reliability by preventing the propagation of errors. Thus, SIGOPS was focused on a very general threat: any computer process, whether accidentally or intentionally created, that could interfere with other legitimately running processes. Butler Lampson, a leader in SIGOPS, recalls this focus on protection: "we didn't so much think of it as security . . . " Instead, "the motivation there [in the time-sharing community] was to keep programs out of each others' hair, not keep data in."[39] Lampson, who earned his Ph.D. in electrical engineering and computer science while working to develop time-sharing systems at UC Berkeley, helped to establish Xerox Palo Alto Research Center (PARC) in 1970. His contributions to computer security were part of a career that has been celebrated for its influence on personal computing and office automation—both of which were pushed primarily by the private sector.[40]

At the 1971 Princeton Symposium on Information Sciences and Systems, Lampson took a very broad view of "protection," defining it as "all the mechanisms which control the access of a program to other things in the system."[41] He explained that

37. R. Graham. 1967. Protection in an information utility. In *Proc. of the First ACM Symposium on Operating System Principles*, 1.1.

38. Ibid., 1.2.

39. Oral history interview with Butler Lampson. December 11, 2014. Charles Babbage Institute. Retrieved from the University of Minnesota Digital Conservancy. Available at conservancy.umn.edu/handle/11299/169983, pp. 10, 32.

40. Lampson won the 1992 Turing award for "contributions to the development of distributed, personal computing environments and the technology for their implementation." Available at http://amturing.acm.org/award_winners/lampson_1142421.cfm.

41. B. Lampson. 1974. Protection. *Operat. Syst. Rev.*, 8(1): 18. Reprinted from B. Lampson. 1971. Protection. In *Proc. Fifth Princeton Symposium on Information Sciences and Systems*, pp. 437–443. Princeton, NJ.

the "original motivation" for protection "was to keep one user's malice or error from harming other users" in any of three ways: by violating the confidentiality of their data, by modifying or destroying their data, or by hogging resources that other users were entitled to use.[42] However, Lampson also noted that protection also had implications for fault tolerance, explaining that these three "reasons for wanting protection are just as strong if the word 'user' is replaced by 'program'," because the same mechanisms could "guarantee that errors in one module will not affect another one."[43] This would increase the reliability of large systems, while also protecting proprietary programs.

Lampson's 1971 paper, which provided a widely-used heuristic for understanding security and protection in operating systems, was inspired by his work on an ACM committee for undergraduate computer science education. He proposed that the committee introduce the notion of a function that would determine what subjects in a computer system had access to what objects: "You feed in the subject and the object and it says yes or no, or maybe it says read or write, or whatever."[44] However, the committee thought this was too confusing, so Lampson proposed something simpler: a matrix in which rows were labeled by particular subjects, and columns were labeled with the objects to which those subjects might have access. Entries in the matrix would then specify what privileges each subject had to each object. Because such a matrix would be sparse and occupy large amounts of memory, it was presented as a heuristic rather than a practical mechanism for access control. Lampson's 1971 paper was so widely discussed that the SIGOPS newsletter, *Operating Systems Review*, published it in 1974.

Lampson's interest in protection was typical of many SIGOPS researchers, who were primarily focused on a vision of a new time-shared computing industry. For example, Dorothy Denning, who was a doctoral student at Purdue University when Lampson presented his work on protection, recalls that she envisioned "a future where you would be using tax preparation software and you would be submitting your data to a service, probably running on some time sharing computer, and how could you make sure that the sensitive information that you provided did

42. Ibid.

43. Ibid.

44. Oral history interview with Butler Lampson. December 11, 2014. Charles Babbage Institute. Retrieved from the University of Minnesota Digital Conservancy. Available at conservancy.umn.edu/handle/11299/169983, p. 60. Hereafter: Lampson oral history.

not get into the hands of people that shouldn't have it."[45] This vision prompted her to develop "a lattice model of secure information flow" for her 1975 doctoral dissertation, which provided a general way of specifying the security mechanisms of an operating system. Denning presented the lattice model at the 1975 Symposium on Operating Systems before publishing it in *Commununications of the ACM*.

Nonetheless, some operating systems researchers were eventually influenced by the concerns of the U.S. Defense Department. Lampson recalled that for the Defense Department, the "threat model is that bad software can get its hands on your data," which was "not something that anyone had ever thought about before these DoD-influenced guys started to get into it."[46] After Lampson learned about these concerns, he decided to make them more publicly accessible by writing "A note on the confinement problem," which was published in *Communications of the ACM* in 1973. Interestingly, Lampson's paper reframed the confinement problem in the language of the private sector, describing the problem created when one program, the "customer," uses another potentially untrustworthy program, the "service." How could the customer ensure that the service did not leak confidential information? Lampson identified three different kinds of "channels" for leaking such information, and identified principles for blocking them.[47]

However, by the late 1970s, researchers shifted from focusing on protection as something that *enhanced* system reliability while preventing misuse, to focusing on the *tensions* between security and reliability. In January 1977, SIGOPS Review presented a summary of a recent workshop on "New Directions in Operating Systems" in which Lampson co-authored a paper on "Protection, Security, and Reliability." The paper noted the need to "Explore the relationship between reliability and protection":

> Is there a danger of the redundancy implicit in fault tolerance compromising security? The objectives of the two studies conflict in that Reliability has to do with making systems continue although they are broken and Protection has to do with making systems refuse totally to do anything which might be unsafe.[48]

45. Oral history interview with Dorothy E. Denning. April 11, 2013. Charles Babbage Institute. Retrieved from the University of Minnesota Digital Conservancy. Available at purl.umn.edu/156519, p. 15. Hereafter: Denning oral history.

46. Lampson oral history, 33.

47. B. Lampson. 1973. A note on the confinement problem. *Commun. ACM*, 16(10): 613–615.

48. B. Lampson, et al. 1977. Protection, security, reliability. *Operat. Syst. Rev.*, 11(1): 13.

The tradeoff between security and reliability was significant because security was not a primary goal in the design of operating systems. A 1973 *Operating Systems Review* article placed "data security" in a list of secondary design goals, whereas system efficiency, adaptability, reliability, and generality of application comprised the list of primary goals.[49] The review went on to evaluate conflicts in design goals, showing that security conflicted not only with all four of the primary goals, but also with secondary goals such as maintainability and compatibility.[50] Thus, SIGOPS publications and forums tended to frame security in the context of a much broader research agenda about improving operating systems.

12.1.4 Mechanizing and Standardizing Security Policies

While members of SIGOPS recognized that computer security had public policy dimensions, they focused instead operating systems policy—that is, rules which were inscribed in operating systems' hardware and software, and thereby controlled access to resources in a mechanized, automated way. They self-consciously inscribed policies from the social world into the mechanized world of the computer, and sometimes justified those policies by reference to "normal" policies in social organizations.

For example, Lampson proposed preventing information leakage through "legitimate channels," such as the bill a confined service might send to a customer, by the principle of "masking," i.e., requiring that the caller determine all of the inputs to such channels. He further noted that while it might initially seem "absurd to allow the customer to determine the bill," this was not the case: "since the service has the right to reject the call, this scheme is an exact model of the purchase order system used for industrial procurement."[51] Similarly, the summary of a discussion of a paper on protection rings at the third Symposium on Operating Systems Principles noted that the "mutual suspicion" between programs which formed the basis for

49. D. H. Abernathy, et al. 1973. Survey of design goals for operating systems. *Operat. Syst. Rev.*, 7(2): 19–34. Similar tradeoffs were noted earlier. In 1973, Lampson noted that security principles such as "enforcement" could impose significant penalties on operating system performance (B. Lampson, A note on the confinement problem). Similarly, in 1975 Lipner noted that "closing the covert channel seems to impose a direct and unreasonable performance penalty" (S. B. Lipner. 1975. A comment on the confinement problem. *Symposium on Operating System Principles*, 192–196).

50. Abernathy, et al., Survey of design goals for operating systems.

51. Lampson, A note on the confinement problem, 615.

the paper "was questioned with the observation that no *human* organization could operate that way."[52]

By mechanizing security policies, operating systems researchers implicitly gave computer manufacturers the primary responsibility for computer security. This view was articulated explicitly at a 1972 workshop on controlled accessibility that ACM co-sponsored with the National Bureau of Standards. A working group chaired by Clark Weissman of System Development Corporation argued that security was "an industry-wide problem not restricted to any one segment," but also expressed a consensus "that the manufacturer has ultimate responsibility for delivering systems that can be operated securely." The group stated that computer facilities managers were responsible for specifying security requirements, vetting products before purchasing them, and operating them in a secure manner, but emphasized the need for techniques to certify the security of new operating systems: "The manager today gets mostly 'arm waving' from the vendor. Government should play an important role in this arena, possibly paralleling its role in commercial aviation, in which the FAA certifies aircraft as airworthy. Alternatively, 'secureworthiness' might be granted by an organization similar to the Underwriters Laboratory."[53]

Although the ACM/NBS workshop was sparked by legislative mandates for NBS to develop privacy standards for federal agencies, it was the Defense Department that took the lead in developing "secureworthiness" ratings, in the form of the Trustworthy Computer System Evaluation Criteria (TCSEC).[54] This led to security policies that reflected the needs of the Defense Department rather than those of the private sector. By the mid-1980s, computer security researchers recognized this explicitly:

> Most discussions of computer security focus on control of disclosure. In particular, the U.S. Department of Defense has developed a set of criteria for computer mechanisms to provide control of classified information. However, for that core

52. M. Schroeder. 1972. Discussion of 'A hardware architecture for implementing protection rings.' *Operat. Syst. Rev.*, 6(1): 166.

53. S. K. Reed and D. K. Branstad, editors. 1974. *Controlled Accessibility Workshop Report: A Report of the NBS/ACM Workshop on Controlled Accessibility, December 10–13, 1972, Santa Fe, California*, p. 21. Government Printing Office, Washington, DC.

54. For discussion of the TCSEC see J. Yost. 2007. A history of computer security standards. In K. De Leeuw and J. A. Bergstra, editors. *The History of Information Security: A Comprehensive Handbook*, pp. 595–621. Elsevier, Amsterdam; D. MacKenzie. 2001. *Mechanizing Proof: Computing, Risk, and Trust*, pp. 177–181. MIT Press, Cambridge, MA.

> of data processing concerned with business operation and control of assets, the primary security concern is data integrity.[55]

Because mechanized policies were only useful if they were implemented correctly, computer scientists sought to formally specify policies and prove their correctness. The goal of formally verifying that operating systems enforced security policies became central to the TCSEC.[56] However, before the TCSEC began developing in 1978, SIGOPS demonstrated a consistent interest in formal verification; at least one-third of SIGOPS papers published from 1975–1977 discussed formal verification.[57]

Members of SIGOPS acknowledged that formally specifying and verifying the correct implementation of security policies would not guarantee security. The fundamental problem was that correctness depended on qualities outside of the machine. For example, a letter published by SIGOPS in January 1978 noted:

> Any logical statement about security is essentially relational and of the form: Given conditions C, user U has access rights A to information I. No practical computer manager is under any illusion that C can ever be completely specified since human relationships and other factors quite outside the logic of the computer system are very relevant to the level of security that is attained. For this reason those who seek to certify system security in a rigorous fashion are ultimately doomed to failure.[58]

This computer security researcher argued that the "proper question to address to a security system is not whether it is correct but whether it is appropriate."[59]

---

55. D. Clark and D. Wilson. 1987. A comparison of commercial and military computer security policies. *Proc. of the 1987 IEEE Symposium on Security and Privacy*, pp. 184–194. Oakland, CA.

56. MacKenzie, *Mechanizing Proof*; Yost, A history of computer security standards.

57. This is based on a keyword search of the titles and abstracts, and corresponds to 6 of 17 papers. Additionally, one advantage of Denning's lattice model (which was not included in this count) was that it provided a way to construct "automatic program certification mechanisms for verifying the secure flow of information through a program" (D. E. Denning. 1976. A lattice model of secure information flow. *Commun. ACM*, 19(5): 236).

58. A. L. Wilkinson. 1978. Letters. *Operat. Syst. Rev.*, 12(1): 4.

59. See also L. H. Hin, A. L. Wilkinson, D. H. Anderson, D. P. Chang, W. Wright, A. J. Mayo, I. T. Viney, and R. Williams. 1981. A penetration analysis of a Burroughs large system. *Operat. Syst. Rev.*, 15(1): 14–25.

12.1.5 Beyond Operating Systems

While SIGOPS focused on formally verifying operating systems policy, other groups focused on different methods. For example, databases raised unique security questions. ACM's SIGs on Management of Data (SIGMOD), Information Retrieval (SIGIR), and Business Data Processing (SIGBDP) co-sponsored the first international conference on Very Large Data Bases (VLDB) in 1975, and helped to establish a new journal, *Transactions on Database Systems* (TODS), which was first published in 1976.[60] A survey of the first three years of TODS shows that "security and access control" was the second most published of 14 areas, second only to "data structures."[61]

Groups concerned with database security recognized that controlling access to individual elements of data would not prevent a database user from inferring information by conducting several carefully designed searches. This problem of inference in databases was analogous to the challenges of controlling information flow in operating systems, and researchers like Dorothy Denning made major contributions in both areas. For example, in a 1979 article published in TODS, Dorothy Denning and two colleagues showed that the problem of the tracker—a statistical formula which a questioner could use to infer information based on several searches—was not unique to queries of small datasets. The conclusion was "that statistical databases are almost always subject to compromise," and that "Severe restrictions on allowable query set sizes will render the database useless as a source of statistical information but will not secure the confidential records."[62] However, the following year, Denning proposed random sample queries as a method for preventing users from inferring confidential data "by making it impossible for a questioner to control precisely the formation of query sets."[63]

Groups interested in database security also focused more than SIGOPS on encryption. For example, in a 1981 paper, cryptographer and University of Wisconsin-Milwaukee professor George Davida and two colleagues explained that physical access controls were useless for preventing access through remote terminals, while

60. R. S. Gaines and D. K. Hsiao. 1975. Preface. *Proc. of the 1st International Conference on Very Large Data Bases*, p. iii.

61. D. K. Hsiao. 1980. TODS–The first three years (1976–1978). *Trans. Database Syst.*, 5(4): 385–403.

62. D. E. Denning and P. Denning. 1979. The tracker: A threat to statistical database security. *Trans. Database Syst.*, 4(1): 76–96 on 77.

63. D. E. Denning. 1980. Secure statistical databases with random sample queries. *Trans. Database Syst.*, 5(3): 291–315.

operating systems security controls inevitably contained vulnerabilities, such as software errors and Trojan horses. In their view, databases were vulnerable because of a "fundamental fact. . . . The DBMS [Database management system] knows what data it is manipulating!"

> Even if users can be effectively prevented from obtaining the data through legitimate channels, the DBMS can leak raw data by omission or subversion, or the storage media can be physically stolen. Also, since the raw data are stored in readable form, spurious entries or updates can be made by anyone penetrating the DBMS.

Cryptography could help to eliminate the problem of disclosure, Davida explained, because "obtaining the data will be of no advantage to a person without the proper key to decrypt it," and "If the DBMS does not know the value of the data, Trojan horse code cannot pass the values."[64]

SIGCOMM focused even more than SIGMOD on encryption, particularly as a solution to the problem of authentication—that is, ensuring that a computer user was who he or she claimed to be, and that data was provided by its apparent source.[65] Rein Turn, a researcher at RAND and leader in ACM's SIGOPS, explained in *SIGCOMM Review* that authentication could be provided by something a user owns, possesses, or knows, but it was essential to securely store and transmit the information used to authenticate the user or process.[66]

12.1.6 SIGSAC

Groups such as SIGCAS, SIGOPS, SIGMOD, and SIGCOMM all treated computer security as one part of a broader agenda. However, in the early 1980s, a new SIG on Security, Audit and Control (SIGSAC) formed to focus primarily on computer security. In principle, SIGSAC might have bridged different framings of computer security, but in practice it introduced its own way of framing computer security,

---

64. G. Davida, D. L. Wells, and J. B. Kam. 1981. A database encryption system with subkeys. *Trans. Database Syst.*, 6(2): 314.

65. Out of 39 SIGOPS papers related to security and published during 1967–1980, only two mention authentication in the title or abstract and only one mentions encryption. By contrast 4 of 17 papers published in SIGCOMM between 1973–1980 mention authentication, and 4 mention encryption. Similarly, 2 of 29 papers published in SIGMOD 1971–1980 mention authentication, while 3 mention encryption.

66. R. Turn. 1975. Security problems in computer communication systems. *Comput. Commun. Rev.*, 5(1): 35–44.

one that focused on the problem of computer crime and managerial solutions to that problem.

In 1981, SIGSAC's first chair, Leslie Ball, noted in the inaugural SIGSAC newsletter: "Hardly a week goes by that we do not see some news story in a national or local newspaper or magazine about some computer crime."[67] Ball was Professor of Information Systems at Babson College, a business school, and other early leaders in SIGSAC had similar credentials. The first SIGSAC Secretary, Martin Bariff, was faculty in the Weatherhead School of Management at Case Western Reserve University, as well as a leader in the Electronic Data Processing Auditors Foundation. SIGSAC built on the traditions of such auditing groups, which were concerned about the unique challenges posed by computerization and a "vanishing paper trail."[68]

SIGSAC published discussions of a variety of methods for securing systems, but from the perspective of management rather than research. For example, SIGSAC demonstrated some interest in encryption standards; the cryptographer Tom Berson was an early leader in SIGSAC, and the group's second business meeting featured cryptographer Leonard Adelman, who was developing cryptographic technology for commercial use—but articles focused on encryption standards and policies rather than the development of novel cryptographic techniques.[69] In other words, SIGSAC focused on publishing information about standards that managers could apply to their computer facilities.[70]

SIGSAC struggled with a relatively inactive membership at first. In 1984, then-chair Jim Schweitzer, of Xerox Corporation, confronted the membership bluntly, proposing three possible reasons for the existence of SIGSAC:

1. To meet, and exchange ideas with, other professionals with similar interests. This reason must not apply to the members of SIGSAC, by and large,

67. L. Ball. 1981. From the chairman. *SIGSAC Rev.*, 1(1): 3.

68. R. E. Anderson. 1981. EDP auditing in the 1980s: Or the vanishing paper trail. *SIGSAC Rev.*, 1(1): 6–15. Reprinted from the Computer Performance Evaluation Users Group, 16th Meeting, National Bureau of Standards Special Publication 500–65, Dr. Harold Joseph Highland, editor, 1980.

69. For more on encryption see Cryptography rumors. 1982. *SIGSAC Rev.*, 1(2): 4. E. Fenna. 1984. Data encryption protocols for electronic mail. *SIGSAC Rev.*, 3(2–4): 43–47.

70. For example, one article suggested the need to develop "a security rating scheme for networks" as "an invaluable guide to EDP managers . . . " (R. S Sharma. 1986. Data communications and security. *SIGSAC Rev.*, 4(1): 36. For early articles that were focused on the development of standards for managers, see M. Bariff. 1981. DP control and standards activities. *SIGSAC Rev.*, 1(1): 20–25. K. A. Forcht. 1986. The need for including data security topics in the college business curriculum. *SIGSAC Rev.*, 4(3): 9–11.

since I heard such resounding silence in reply to my letters [suggesting local meetings among members].

2. To receive an interesting journal of professional information. . . . No one submits any articles to the REVIEW. We have to scrape and scratch, and the results of such meagre product are to be seen in the quality of our publication. So, this doesn't seem to be the reason, either.
3. To contribute funds to the overhead costs of ACM. In my opinion, a stupid reason to belong to a SIG. Is this actually our purpose?[71]

Despite Schweitzer's appeal for more contributions from members, he and SIGSAC leaders continued to struggle. Lance Hoffman, a computer science professor at George Washington University, recalls being a member of SIGSAC, but noted that it "For some reason, it didn't get the legs behind it" that other groups, such as the IEEE Symposium on Security and Privacy did.[72] Tom Berson was disappointed by the group, recalling: "They said oh, we're going to have a SIG so I joined it and then nothing ever happened."[73]

By the late 1980s, SIGSAC began to see more contributions to the newsletter. For example, SIGSAC provided an outlet for researchers who were skeptical about formally specifying and verifying the correctness of operating systems. A satirical article by Bill Neugent proposed several "preposterous opinions about computer security," including that "formal specifications are overdressed." The author described his experience with the "lowly responsibility of formulating security test ideas for a system," explaining:

> I say lowly because the system in question was so rich and fortunate that it was being ministered to by an entire troupe of specialists in formal specification and verification. To me it was a tremendous honor simply to be associated with a system that had been touched by such remarkable illuminati . . .
>
> In four months, with help from several colleagues, I had a stack of outrageous things [security tests] to try. On trying them, I was astonished to discover a significant number of ways that users at terminals could romp all over that system. This came as quite a surprise to everyone, especially the program manager, who kept shaking his head and saying "But it's been proven."

71. J. Schweitzer. 1984. Message from the chair. *SIGSAC Rev.*, 3(1): 2.

72. Oral history interview with Lance Hoffman by Rebecca Slayton. July 1, 2014. Charles Babbage Institute. Retrieved from the University of Minnesota Digital Conservancy. Available at http://hdl.handle.net/11299/168279, p. 61.

73. T. Berson. Interview with author. April 18, 2014. Palo Alto, CA.

He continued: "Now don't get me wrong. I'm not saying that this one example absolutely proves formal specification and verification to be ineffective. Obviously for that I'd need either (1) an FTLS [formal top level specification] and 20 man years of coverage or (2) at least three beers worth of analysis."[74]

As these words suggest, some computer researchers begrudged the prestige associated with formal verification, and SIGSAC provided a forum for expressing such irreverent views. However, it did not take a central role in encouraging broad discussions of computer security until the 1990s, as we will see below.

12.2 Transitions in the 1980s: Specialization and the Growth of Computer Networking

As the previous sections suggest, ACM SIGs framed computer security in distinctive ways in the 1970s. For most of these groups, computer security was simply part of a broader research agenda, such as operating systems, databases, communications, or social implications of computing. In some ways, the SIGs may have encouraged computer security researchers to pursue single approaches to computer security; SIGCAS, SIGOPS, SIGMOD, and SIGCOMM each focused on a slightly different set of threats, goals, and methods. These different framings were not necessarily contradictory, but they were constructed largely independently. If anything, SIGs tended to encourage researchers to work in silos, each pursuing security as part of a separate research agenda.

This began to change in the 1980s, partly because groups began to specialize in computer security.[75] Perhaps most notably, in 1980, George Davida and Rein Turn helped to initiate the Symposium on Security and Privacy (or Oakland Conference), sponsored by the IEEE Computer Society's Technical Committee on Security and Privacy. Participants in the first SSP recall that there "was initially a strong partition between system security and cryptography," citing the example of how "after a cryptography session at the 1981 SSP, all the panelists left the room for private discussions and missed the subsequent operating system and formal methods panel." The 1982 SSP attempted to overcome divisions between cryptographers and systems researchers with "a panel that approached operating systems, cryptography, and formal methods in a single session." Eventually, conference participants seem to have accepted the adage that "If you think that cryptography is the answer to your

74. B. Neugent. 1986. Preposterous opinions about computer security. *SIGSAC Rev.*, 4(3): 4–5.

75. For example, in 1979, as part of the Defense Department's Computer Security Initiative, NBS and NSA began to sponsoring a seminar on computer security, which by 1985 had become an annual National Computer Security Conference.

problem, then you do not understand cryptography and you don't understand your problem."[76]

Some computer scientists argued against specializing in computer security. In 2014 Lampson recalled: "I never thought of myself as a specialist in computer security. In fact, I don't believe in computer security as a specialization. . . . it's a branch of computer systems, and the people who specialize in it are not as good as the people who do it as a sideline."[77] As a result of this philosophy, Lampson did not generally attend computer security conferences unless he was invited to speak. However, he remained active in SIGOPS, and to date, three of his papers have won the SIGOPS Hall of Fame Award.

By contrast, Dorothy Denning became very involved in the IEEE Symposium on Security and Privacy (SSP) once it began in 1980, and eventually took a leading role in helping establish the ACM-sponsored Computers, Freedom, and Privacy conferences in the early 1990s (discussed below). But with increased involvement in the computer security community came less involvement in SIGOPS; Denning served in leadership roles in the 1970s and early 1980s, but recalls that she did not stay very active in SIGOPS.[78]

In the 1980s, even as groups began to specialize in computer security, changing technologies and new applications of computers sparked debate about the appropriate ways to frame computer security. The widespread application of microprocessors shifted visions of a centralized computer utility toward distributed computing, in which many users possessed machines which could process information independently and communicate with one another. Computer networking grew rapidly, whether it was electronic funds transfers between banks, e-mail exchanges between individuals with personal computers, or communications in safety-critical process control systems such as nuclear power plants.

The increased use of computer networking called attention to shortcomings in formal specification and verification of security policies. As early as 1975, at a workshop that was co-sponsored by SIGOPS and SIGCOMM, Stockton Gaines chaired a session on protection which concluded that the confinement problem "appears to be a very hard problem to solve and in the context of a network it may not

76. P. G. Neumann, S. Peisert, and M. Schaefer. 2014. The IEEE Symposium on Security and Privacy, in retrospect. *IEEE Secur. Priv.*, 12(3): 3.

77. Lampson oral history, 32.

78. Denning oral history.

[sic] be a very severe problem."[79] By contrast, at a July 1980 workshop, Dave Clark of MIT discussed potential advantages of distributed computing, including the idea that "Distributed systems can provide more security than a centralized system." According to the workshop summary published in *SIGOPS Review*, this idea "was hotly debated and produced the statement that security results from a logically distributed system, and not from physical distribution." Nonetheless, workshop participants noted that "physically distributed systems, where your part resides at your site, provide more apparent protection of information by providing physical control over that part of the system; information can flow in and out only over easily identified wires. This produces 'warm feelings' in the user of the system."[80]

In October 1980, *SIGOPS Review* published an exchange of views in which Harold Stone, a computer science professor at the University of Massachusetts, proposed that "protection is simply achievable through personal or dedicated computers." Peter Denning disagreed: "I am unconvinced that the issues of protection are fundamentally different in distributed systems as they are in centralized systems."[81]

While the challenges of protection did not disappear with distributed systems, they changed in ways that demonstrated limitations in the approach that SIGOPS had nurtured, and that was institutionalized in the Defense Department: formally specifying security policies and verifying their correct implementation. By the late 1980s, security researchers showed that even if individual operating systems were proven to be secure individually, they were no longer secure when connected together.[82]

12.3 Reframing Security Amid Growing Computer Networking

12.3.1 New Risks: Critical Infrastructure

Increased networking raised questions not only about appropriate methods for securing systems, but also about the risks of insecure computer systems. In 1977,

79. R. S. Gaines. 1975. Session 3: Protection, in *Proc. of the ACM SIGCOMM/SIGOPS Interprocess Communication Workshop. Operat. Syst. Rev.*, 9(3): 58.

80. J. L. Peterson. 1975. Notes on a workshop on distributed computing. *Operat. Syst. Rev.*, 13(3): 20.

81. P. Denning and H. S. Stone. 1980. An exchange of views on operating systems courses. *Operat. Syst. Rev.*, 14(4): 80, 82.

82. D. McCullough. 1988. Noninterference and the composibility of security properties. *Symposium on Security and Privacy*, 177–186.

the Swedish Ministry of Defense conducted a study which identified 16 "vulnerability factors" associated with dependence on computers, and concluded that Sweden's computer dependency was dangerous. Prompted partly by this study, the international Organisation for Economic Co-operation and Development (OECD) organized a workshop on "computer vulnerabilities of society" which was attended by delegations from 16 nations, including the U.S.

In 1980, AFIPS responded to these growing concerns with a study of computer-related vulnerabilities in the U.S. The study was chaired by Rein Turn and included other ACM leaders such as David Brandin, Donn Parker, and Willis Ware. Although concerns about dependence on computers were not triggered solely by the rise of computer networking, the growth of complex networks created unique vulnerabilities:

> Increasingly, stand-alone computer systems have been replaced by systems connected by data communications networks. Many systems are international: they serve multinational corporations or offer data processing services to customers around the world. . . . new weaknesses arise from the increase in complexity (larger systems with more interconnections) and from the resource-sharing nature of their operations (processors, data, and communication channels shared by the many users).

The AFIPS study examined the same 16 issues identified in the Swedish report, which included well-established risks such as computer-related crime, or the inappropriate or harmful use of personal information in databases. However, the report also noted a new risk, "functionally sensitive systems." As the report explained:

> Many computer-based systems must be regarded as being functionally critical since they are used in societally critical activities such as transportation, process control, electrical power distribution, telecommunications, distribution of goods and services, banking, and law enforcement. The possible loss of the use of an entire functional area is of major concern.

Although process control computers had been in use since the 1960s, the group expressed concern about the fact that in "modern, networked computer systems it is possible for incorrect data to be disseminated throughout the system or to deadlock inoperative all computers and/or databases in the system. Such events may require weeks for total recovery . . . " Significantly, the study did not consider the possibility of hackers or foreign adversaries disabling or destroying critical infrastructure; it would be over a decade before these concerns became prevalent.[83]

Overall, the AFIPS study concluded that the "United States, due to its size, decentralization, and redundant facilities can absorb the shocks of everything but a large-scale concerted hostile action." However, the group also noted that this could change, and placed responsibility for maintaining information resiliency in "in all sectors of society—the general public, the government, and the information industry." It called for greater public awareness, and for the government to establish standards, but concluded:

> . . . most of the responsibility for information resiliency lies in the private sector. User organizations, as well as providers of information systems or services, should decide on reasonable protective measures and be willing to pay for them. Backup and recovery plans and exercises, data quality control checks, and data security controls are the best insurance against adverse events. Taken collectively, these actions enhance national information resiliency.[84]

Significantly, AFIPS regarded "acts of war" to be outside the scope of its study, although it noted that electromagnetic pulses from nuclear blasts would destroy computer equipment. Nonetheless, many computer professionals were worried about interactions between computing and war. In the early 1980s, the brash rhetoric of Ronald Reagan's presidential administration—which discussed nuclear war as though it might be winnable—prompted many computer professionals to worry that nuclear weapons depended critically upon computer systems that were far from perfect, and that could even spark an accidental nuclear war in a time of international tensions. Several computer professionals in Silicon Valley formed Computer Professionals for Social Responsibility (CPSR) in an effort to raise awareness about the risks of accidental nuclear war.[85]

In October 1982, CPSR urged the ACM Council to take a stance on computer technology and the risk of nuclear war. As in the 1960s, the ACM refused to make such a potentially "political" statement. But after chartering a study on the risks of safety-critical computing, the ACM Council approved a resolution declaring the "reality" that "computer systems can and do fail," sometimes posing "extreme risk

83. R. Turn, E. J. Novotny, J. J. Geraghty, E. H. Sibley, and W. H. Ware. 1982. *Observations on the Resilience of the U.S. Information Society*, quotes on pp. 3–4, 13–14. American Federation of Information Processing Societies, Arlington, VA.

84. Ibid., 19, 22.

85. See R. Slayton. 2013. *Arguments That Count: Physics, Computing, and Missile Defense, 1949–2012*. MIT Press, Cambridge, MA.

to the public." They warned: "Increasingly, human lives depend upon the reliable operation of systems such as air traffic and high-speed ground transportation control systems, military weapons delivery and defense systems, and health care delivery and diagnostic systems."[86]

The ACM Council also tasked the Committee on Computers and Public Policy (CCPP) with a study of computer system reliability and risks. Peter Neumann, a computer security researcher at SRI and the chair of the CCPP, partly responded to his new charge by establishing a moderated e-mail forum, "RISKS". Although they were initially prompted in part by concerns about military weapons systems, both the Risks Forum and CPSR soon became involved in a tangled debate over privacy, crime, and national security.

12.3.2 The Shifting Privacy Debate

In addition to calling attention to new risks, increased computer networking brought together and transformed long-standing debates about privacy and national security. The 1970s privacy debate focused on the threats that corporate and government record-keeping organizations posed to individual consumers and citizens. However, in the 1980s, organizations and individuals began to use the same computer-mediated communications, such as e-mail, making both more vulnerable to intrusions. Furthermore, a key solution to this new vulnerability—the widespread use of encryption—also came to be seen by some as a threat to national security.

In the 1970s, Ware frequently testified before Congress about the need for record-keeping organizations to protect the privacy of individuals whose information was stored, but in the 1980s his focus shifted to the risk shared by both individuals and businesses who used computer-mediated communications. In 1983, Ware warned Congress that the privacy problem could no longer be interpreted primarily in terms of "record-keeping processes," because the "widespread application of computer and communication systems to provide a broad spectrum of services will put on the horizon many new dimensions of the privacy problem." As an example, he explained that electronic mail represented a "comprehensive business records service" which was "much more efficient than writing letters, making phone calls, and then writing memoranda-of-record." He continued: "I can organize the messages by folders and subfolders so that the system becomes a com-

86. A. Goldberg. 1985. Reliability of computer systems and risks to the public. *Commun. ACM*, 28(2): 131–133.

prehensive automated filing and retrieval system. Anyone having access to this body of information might just as well have the key to my office and to its file cabinets."

> Such a comprehensive business records service is really what electronic mail is all about, and it is a service that will soon be offered by the private sector. Can you imagine the possibilities for various degrees of wrongdoing and snooping when all that information—both private and corporate—gets into electronic mail systems? Can you imagine what a lucrative target it will become for all sorts of reasons?

Ware next raised questions of responsibility and liability:

> What are the vendor's obligations in terms of providing comprehensive security safeguards for his system? Should they be mandated by law? Should it be *caveat emptor* for both private sector and government alike? Should the government be concerned that so much corporate information is subject to penetration by unfriendly agents?

As this suggests, Ware now viewed corporations and private sector organizations as vulnerable to the same kinds of intrusions as individual citizens. Ware, who had recently served on the advisory panel to the AFIPS study on vulnerabilities in computerized societies, called for a national commission "to examine the possible vulnerabilities of our highly computerized society."[87]

Importantly, Ware noted that computer communications were vulnerable in large part because encryption technologies were not widely available. However, as cryptographers attempted to make strong encryption more widely available to businesses and individuals, the U.S. National Security Agency objected to what it viewed as a threat to its surveillance capabilities. Cryptographers took rapid strides in the late 1970s and 1980s, and ACM was among professional organizations that helped disseminate new research. For example, at the 1976 AFIPS annual meeting, Whitfield Diffie and Martin Hellman noted that "In a computer network with a large number of users, cryptography is often essential for protecting stored or transmitted data." They went on to present "ways to preserve privacy of communication despite the large number of user connections which are possible . . . "[88] Diffie's and Hellman's work, which was also published in IEEE journals, eventually became the basis for encrypting messages in networks with many users (i.e., the Internet).

87. W. H. Ware. 1984. Information systems security and privacy. *Commun. ACM*, 27(4): 318, 320.

88. W. Diffie and M. Hellman. 1976. Multiuser cryptographic techniques. *National Computer Conference*, 109. See also R. Merkle. 1978. Secure communications over insecure channels. *Commun. ACM*, 21(4): 294–299.

However, the NSA objected to publicizing improved security methods. Matters came to a head in 1978, after Davida and his student filed a patent for an encryption device, and the Commerce Department (acting at the behest of the NSA), issued a secrecy order, not only on the device but on the ideas used to develop it—which had been completely unclassified. Davida was not alone in receiving such an order, and although it was eventually lifted, Davida and others went public, raising a firestorm of controversy over the relationships between national security, academic freedom, and privacy.[89]

In 1980, the American Council on Education appointed a study group on public cryptography, which included David Brandin, the Vice President of SRI as a representative of ACM, as well as Davida and Hellman. The group's final report, which was published in *Commununications of the ACM*, framed the problem as a "balance" between national security and freedom of speech. It recognized that nongovernmental research on cryptography might "adversely affect the security of the nation's sensitive official communications and the nation's ability to obtain and understand foreign intelligence." However, it also acknowledged that public cryptography could "increase communication protection in commercial and private fields, thus enhancing the security of private and commercial communications and ultimately furthering the nation's welfare and security in a broader sense."[90] Ultimately, it recommended a system wherein researchers would voluntarily submit their publications to NSA review before publishing them.

However, in a minority opinion, George Davida argued that there was no tension between freedom of speech and national security, broadly defined. He emphasized that cryptography was important to protecting the privacy of databases, operating systems, and communications, for both businesses and the private sector, noting that the "emergence of COMPUTER NETWORKS has led to new applications that threaten privacy to a degree that was not possible before." While acknowledging that publishing work on encryption techniques might create difficulties for the NSA's mission, narrowly construed, he argued that the "NSA can perform its mission the old-fashioned way: STAY AHEAD OF OTHERS."[91]

Despite Davida's views, the ACM Council ultimately voted to endorse the voluntary review system, and to periodically publish notification to potential authors of the review system. However, it also insisted that "ACM considers authors' coopera-

89. See S. Levy. 2001. *Crypto: How the Code Rebels Beat the Government—Saving Privacy in the Digital Age*. Viking, New York.

90. P. J. Denning, D. H. Brandin, D. C. Schwartz, and G. I. Davida. 1981. Report of the public cryptography study group. *Commun. ACM*, 24(7): 434–450, on 438.

91. Ibid., 447, 450.

tion with NSA to be purely voluntary and not a precondition for review or publication by ACM."[92] It also continued to publish work on cryptography, including critiques of the National Bureau of Standards' Data Encryption Standard, which many cryptographers believed had been designed to be weak at the request of the NSA.[93]

12.3.3 The Hacker Debate

While the growth of computer networking and related technologies prompted some debate about individual privacy, most policymakers and reporters focused much more on the threat that individuals posed to private and government organizations—what was commonly called computer crime. While computer crime drew some attention in the 1970s, the growth of computer networking greatly increased opportunities for activities that could be framed as criminal, and associated publicity. However, it also raised questions about what constituted computer-related crime, who was responsible for it, and what solutions were most appropriate.

In the early 1980s, newspapers began publicizing a new generation of computer "hackers," typically young people who used networks to gain unauthorized access to computers. For example, in August 1983, newspapers across the United States covered the story of how teenagers living in Milwaukee successfully penetrated a computer system at Los Alamos National Laboratory in New Mexico.[94] The 1983 film *Wargames* presented a fictional account of a high-school student hacking into the U.S. nuclear command and control center in Colorado, and suggested the high stakes of hacking.[95]

92. Official ACM: ACM council meeting minutes summary. 1982. *Commun. ACM*, 25(12): 955–956.

93. W. Diffie and M. E. Hellman. 1981. On the security of multiple encryption. *Commun. ACM*, 24(7): 465–467. F. Weingarten. 1981. Cryptographic research and the computer science community. *Commun. ACM*, 24(12): 851–853.

94. See J. Treaster. August 23, 1983. Trial and error by intruders led to entry to computers. *New York Times*; AP. August 12, 1983. F.B.I. weighs action in 'raids' by youths on computer data. *New York Times*; D. Burnham. August 13, 1983. Computer security raises questions. *New York Times*; W. J. Broad. August 26, 1983. Rising use of computer networks raises issues of security and law. *New York Times*.

95. In 1983, the House Committee on Small Business opened hearings on a computer crime bill with reference to *Wargames*, while the House Committee on Science and Technology opened hearings on "Computer and Communications Security and Privacy" with a brief clip from *Wargames*. House Committee on Science and Technology. *Computer and Communications Security and Privacy* (Ninety-eighth Congress, First session, September 26, October 17, 24, 1983). House Committee on Small Business, *Small Business Computer Crime Prevention Act* (Ninety-eighth Congress, First session, July 14, 1983).

The U.S. Congress responded to such concerns by debating legislation related to hacking, and eventually passed two separate acts. The Computer Fraud and Abuse Act of 1986 focused on criminalizing particular kinds of activities, while the Computer Security Act of 1987 represented an effort to balance responsibility for computer security between civilian and military government sectors. Although ACM was not officially represented in Congressional hearings on these acts, ACM leaders such as Donn Parker, Willis Ware, Buck Bloombecker, and Tony Oettinger provided perspectives that were well informed by ongoing debates within ACM.[96]

Parker was among ACM members who framed hacking as unambiguously criminal. Although in the 1970s, most of the crime studied by Parker was perpetrated by computer professionals working within their organizations, by the early 1980s his attention increasingly focused on those who used phone lines for long-distance break-ins to computer networks. In 1982 Congressional testimony, he noted that SRI would soon undertake a major study on "these new criminals, the juvenile system hackers."[97]

By contrast, other ACM members argued that the focus on hackers was misguided. One suggested that "in most cases no criminal intent exists" and that the public debate about hackers was a distraction from bigger problems. He noted that "the emotional issue of high school kids intentionally sabotaging computer networks fits in very nicely with the Reagan policy of tightening government controls" over information. He also pointed out that computer managers blamed hackers because "many data centers have poor security, and the people running them know it." [98] In other words, the focus on hackers conveniently shifted responsibility for security away from the manufacturers and managers of computer systems—most of whom did not want to spend money or time on improving technical and procedural security controls.

Another ACM member similarly objected to the idea "that tough laws and prosecution of teenagers is going to solve their [corporations'] security problems." He

96. See, for example, House Judiciary Committee. *1984: Civil Liberties and the National Security State* (Ninety-eighth Congress, First and second sessions, November 2, 3, 1983, and January 24, April 5, and September 26, 1984, 1984).

97. House Judiciary Committee. *Federal Computer Systems Protection Act* (Ninety-seventh Congress, 2nd Session September 23, 1982), 52. By comparison, Parker's 1976 survey of "computer abuse" found that the most common vulnerability was in the manual handling of input and output data, followed by an absence of physical restrictions to the computer facility (D. Parker. 1976. Computer abuse perpetrators and vulnerabilities of computer systems. *National Computer Conference*, 65–73.

98. D. Bellin. 1985. High school hackers: Heroes or criminals? *Comput. Soc.*, 14 & 15: 17.

described his experience working "for a major high-tech U.S. firm that had every reason to worry about losing its secrets."

> I have sat through security awareness briefings, and signed papers saying I understood the sensitive nature of information I worked with and on and that I agreed to not divulge it and to not treat it carelessly. Then I have struggled in frustration to get my superiors to treat seriously the problems of protecting the computer data. I have been told that most problems come from current employees or former employees so you put procedures into place to keep employees off the computer and change the passwords immediately after someone quits. I have also been told that it does not matter what the procedures are, just as long as you look like you have some so that the auditors do not get on your case . . .

He noted that the focus on teenage hackers ignored a greater threat: "Criminals who have plenty of money and plenty of time and employers who are downright encouraging of their criminal behavior (read KGB)." He concluded: "If we are going to toughen up the laws for computer crime we need to be very careful to structure these laws so that the burden of proof remains where it belongs, on the corporation's data processing department (we do not allow self-incrimination in other types of crimes)."[99]

The most radical view on hacking came from Richard Stallman, who argued that hacking "is not a grave problem, but is a response to an underlying problem which is grave: the trigger happy hostility, suspicion, and general bad-neighborliness of the owners of computer installations. Computer security is the tool they use to implement policies of inhospitality."[100] Few ACM members supported this view, and many ridiculed it.

Instead, most of the ACM's discussion of hackers focused on the education and ethics of young people, rather than the responsibilities of computer managers. This focus was clear in a 1985 ACM panel on hacking, which included researchers such as Donn Parker, along with hackers and educators. The panel sought ways "to deal more effectively with the career criminal through the legal system while channeling the teenage hacker's unbounded inquisitiveness into constructive learning and use." To improve state and federal laws, the panel recommended that ACM help define terms such as "computer abuse," develop better forensics for computer-related crime, and identify who should be held accountable for securing computer products. And to direct curious young hackers towards ethical and productive

99. J. Reeves. 1984. ACM forum: Trojan horses and trusty hackers. *Commun. ACM*, 27(11): 1086.

100. R. Stallman. 1984. Hacking away at morality. *Commun. ACM*, 27(1): 8.

activities, the group recommended developing codes of ethics, and providing a wide range of computer and other resources that would provide a better education. In conclusion, the group speculated that "If only a small portion of the funds expected to be expended in improving security and reliability were to be channeled toward education, perhaps a large percentage of the problem could be eliminated."[101]

Some ACM members criticized the panel for being too easy on hackers. One scoffed at the proposal "that we amuse potential electronic scoundrels by giving them computing power free and letting them find out from each other how nasty unethical activity can be."[102] Another member described the "discussion of the empowerment of interested users" as "ridiculous." He asked:

> What percentage of elementary and secondary school students cannot be challenged by tasks in Pascal, Fortran, assembly languages, or even Basic? I believe it to be a small one. User groups provide a source of interesting projects as well as an avenue to social development for students with extra interest in the subject.[103]

Nonetheless, a few ACM members argued that the panel's focus on legislation and the ethics of hackers was misguided, and that the focus should be on technological solutions and the responsibilities of computer managers. One warned: "In addition to not being effective, comprehensive laws against hacking will have bad effects on the field. Expert lawyers will need to be consulted for every decision of importance. The effect on the economy could be severe." Instead, he argued that "technology should be the first line of defense against hacking," and that the panel "should consider the need to educate the people who control electronic information about techniques for protecting the information."[104] Similarly, a member wrote that while "hacker-oriented legislation may deter hackers, it is the serious, professional criminal who is the real threat. We should not allow our romantic notions of the notorious hacker to set up a straw man for legislation that may prove to do more harm than good." Further, he argued, "Organizations that maintain databases of sensitive information should not feel relieved of their obligation to safeguard them with the introduction of punitive legislation."[105]

101. J. A. N. Lee, G. Segal, and R. Steier. 1986. Positive alternatives: A report on an ACM panel on hacking. *Commun. ACM*, 29(4): 297, 299.

102. J. A. Wills. 1986. ACM forum: Reported hacking. *Commun. ACM*, 29(7): 590.

103. R. P. Canavan. 1986. ACM forum: Reported hacking. *Commun. ACM*, 29(7): 590.

104. P. Purdom. 1986. ACM forum: Reported hacking. *Commun. ACM*, 29(7): 588–589.

105. P. Wettersten. 1986. ACM forum: Further observations on hacking. *Commun. ACM*, 29(9): 829.

Nonetheless, when Congress finally passed the Computer Fraud and Abuse Act of 1986, it did not hold organizations responsible for safeguarding computers, but instead focused on criminalizing unauthorized access to computers. The adequacy of such laws became hotly disputed after the release of the first Internet worm.

12.3.4 The Morris Worm

The debate about hacking heated up after November 2, 1988, the day that Robert T. Morris, Jr., a graduate student at Cornell University, released a worm that (contrary to his intentions) wreaked havoc on the Internet. In his January 1989 president's letter, Bryan Kocher called the Morris worm a "hygiene lesson," noting: "Potential invaders of UNIX networks must be heartened to note how easily and frequently security can be breached through Internet."[106] However, in the June 1989 issue of *Communications of the ACM*—which was almost entirely devoted to articles about the worm—some ACM members objected to Kocher's emphasis on technical problems and the associated responsibilities of computer managers, rather than on the moral failings of young people.

For example, Donn Parker wrote that "We must condemn the lack of computer security that facilitates contamination; however, we would be irresponsible if we did not condemn even more vigorously those who irresponsibly create, experiment with, and insert these most dangerous programs into other people's computers."[107] Another reader objected similarly: "In Bryan Kocher's view, the victim is to share the blame of the crime and be taught a lesson by it if somehow he or she was open to attack."[108] Eugene Spafford, a computer scientist at Purdue who played a major role in analyzing the worm, argued: "Perpetrators of 'viruses' and other 'electronic pranks' should be identified as what they are: vandals."[109] In a separate analysis of the worm, Spafford argued against the idea "that the victims of the worm were somehow responsible for the invasion of their machines," calling this "equivalent to blaming an arson victim for the fire because she didn't build her house of fireproof metal."[110]

However, another ACM member quickly objected to this view, acknowledging that "we do not blame arson victims for living in fire traps . . . but we do blame the

106. B. Kocher. 1989. President's letter: A hygiene lesson. *Commun. ACM*, 32(1): 3.

107. D. Parker. 1989. ACM forum: Beyond worms. *Commun. ACM*, 32(6): 673.

108. E. B. Heinlein. 1989. ACM forum: Beyond worms. *Commun. ACM*, 32(6): 674.

109. T. Narten and E. H. Spafford. 1989. ACM forum: Beyond worms. *Commun. ACM*, 32(6): 673–674.

110. E. H. Spafford. 1989. Crisis and aftermath. *Commun. ACM*, 32(6): 685.

manufacturers of such homes, particularly when the flaw involved has been widely known for years."[111] He continued:

> . . . while Spafford praises the efficacy of the "Unix 'old boy' network" in fighting the worm, he does not explain how these self-appointed fire marshals allowed such known hazards to exist for so long. It is true that we cannot expect bug-free software, and that neither Sun, Berkeley, nor the administrators of the Internet intended any harm by their negligence—but Spafford does not extend the defenses of fallibility and good intentions to Morris. If the Internet is entering a new age of accountability, where computers will have the same legal protections as our homes, then Morris is not alone in failing to live up to his responsibilities.[112]

Similarly, M. D. McIlroy of Bell Laboratories objected that the trapdoors that were exploited by the worm had in some cases been put there deliberately, and that Morris was no more guilty of a felony than were the many people who ignored the vulnerabilities: "Those who repudiate only the final misdeed implicitly absolve the prior lapses in ethics and in oversight that made the worm fiasco possible. They signal that you can get away with building, promoting, distributing, and selling deceitful software as long as you are not caught exploiting it."[113]

Dispute continued after January 1990, when Morris was the first person convicted of a felony under the 1986 Federal Computer Abuse Act. Some felt that the punishment was insufficient. The May 1990 *Commununications of the ACM* news track reported that Spafford was urging his colleagues to refuse to do business with any company that employed a hacker, arguing: "This is like having a known arsonist install a fire alarm. Just because he knows how to set a fire doesn't mean he knows how to extinguish one."[114]

Peter Denning objected to this view in an editorial: "hackers do not constitute an identifiable or cohesive community and employers ought to be free to take risks on whom they hire; moreover, it makes a false analogy between someone who intends to cause damage and Morris, who did not." However, Denning also noted that Spafford was "not a lone advocate of this position. Others have asked the ACM officers to endorse it, and thus far, to their credit, the ACM officers have declined." Denning had the "impression that most computer people feel Morris's sentence

111. J. Matthews. 1989. ACM forum: Worm turns. *Commun. ACM*, 32(9): 1044.

112. Ibid.

113. M. D. McIlroy. 1990. ACM forum: Green light for bad software. *Commun. ACM*, 33(5): 479.

114. R. Steier. 1990. News track. *Commun. ACM*, 33(5): 477.

was appropriate," but also noted that a "sizable minority say the penalty was not severe enough: a jail term should have been included." Denning objected that the logic of such a penalty—sending a signal to somebody who might be considering computer crime—was unsuitable punishment for someone who did not intend to do wrong.[115]

The sentencing of Morris was of deep concern to computer professionals because legislators were considering passing new laws that would impact computer workers. In his June 1989 President's letter, Kocher reviewed several well-publicized incidents of computer crime or hacker activity and took note of state and federal legislative responses. He asserted: "If we, the leaders of the computing profession, do nothing, there will soon be a crazy quilt of state laws regulating the computing profession in fifty different ways." He argued: "It is time for ACM, in cooperation with IEEE and others if possible, to propose and strive for adoption of appropriate federal standards for the computing professions."[116]

Kocher's proposal was supported by some ACM leaders, such as Bloombecker, then chair of ACM's legal issues committee.[117] However, other ACM members reacted negatively to Kocher's suggestion. One objected to any legislation "specifically aimed at computing professionals," because "laws should apply equally to everyone." He concluded that "the last thing we need is to invite government into our profession to regulate us 'for our own good'."[118] Another reader wrote in agreement:

> There will always be someone lurking in the shadows who will attempt to justify bureaucratic control over selected professions. Our obligation as members of the ACM is to oppose any suggestion of governmental regulation, regardless of the source. We have an obligation, however, to start at our own front door. We must each embrace the principles of individual responsibility.[119]

As this suggests, some ACM members preferred individualistic and ethical rather than legislative solutions to the problems of computer security.

Nonetheless, in the early 1990s ACM became more involved in efforts to improve legislation related to computer security. Although ACM had no official representation in the 1989 and 1990 Congressional hearings about amendments to the

---

115. P. Denning. 1990. Editorial: Sending a signal. *Commun. ACM*, 33(8): 11–13.

116. B. Kocher. 1989. President's letter. *Commun. ACM*, 32(6): 661, 662.

117. D. Crawford. 1989. From Washington: Two bills equal forewarning. *Commun. ACM*, 32(7): 780–782, on 782.

118. R. Schwarz. 1989. On regulation and legislation. *Commun. ACM*, 32(8): 918.

119. J. W. Weaver. 1989. ACM forum: More on regulation. *Commun. ACM*, 32(11): 1284.

Computer Fraud and Abuse Act, Marc Rotenberg, the director of the Washington office of Computer Professionals for Social Responsibility, testified and cited considerable material that had been published in the *Commununications of the ACM*.[120] And after law enforcement agencies began a "hacker crackdown," ACM leaders established a forum for discussing new questions emerging around the Internet.[121]

12.3.5 Establishing the Conference on Computers, Freedom, and Privacy

In 1990, Dorothy Denning was working on an article about the culture of hackers when she received a call from Craig Neidorf, a college student and co-publisher of the electronic newsletter *Phrack*, requesting a copy of her paper. Neidorf had been recently indicted by a grand jury on multiple felony counts for electronically publishing a text file, E911. According to federal prosecutors, the file was a highly sensitive document that had been stolen from BellSouth for purposes of illegally manipulating the 911 system. However, Denning "did not see how the E911 file could be used to break into the 911 system or, for that matter, any computer system." She was concerned that Neidorf had been wrongly indicted, and "that a wrongful conviction—a distinct possibility in a highly technical trial—could have a negative impact on electronic publication."[122] Denning helped the defense lawyers prepare their cross-examination of the government's witnesses, and planned to testify during the second week of the trial. However, four days after the trial began, the government dropped the charges.

The Neidorf case occasioned a debate in *Commununications of the ACM*. Denning acknowledged need to punish bad behavior, but also insisted that "system administrators who permit lax security are culpable for their own negligence," and suggested that the severity of a crime be judged in part by whether or not adequate security had been provided. Donn Parker argued that Denning's account was "biased" and presented the hackers' work euphemistically. He also expressed his view that "determining if an offense is a crime depends on the intent of the perpetrator and the seriousness of the act, and not on the vulnerability of the victim."

120. Senate Judiciary Committee 101rst Congress second session. "Computer Abuse Amendments Act of 1990; Hearings Held July 31 1990". House Judiciary Committee 101rst Congress first session. "Computer Virus Legislation; Hearings held November 8, 1989" (1989).

121. See B. Sterling. 1992. *The Hacker Crackdown*. Available at www.gutenberg.org/ebooks/101.

122. D. E. Denning. 1991. The United States vs. Craig Neidorf: A debate on electronic publishing, Constitutional rights, and hacking. *Commun. ACM*, 34(3): 26, 35.

Eugene Spafford presented similar objections, though he also recognized that insurance rates might reasonably go up on someone who didn't properly protect their property.[123]

Neidorf's case was one of many that raised questions about freedom of speech and civil liberties on the Internet. In 1990, Jim Warren, a computer programmer and consultant who had held leadership positions in ACM, and others responded by organizing a Conference on Computers, Freedom and Privacy (CFP) sponsored by CPSR, ACM, IEEE, and several other groups.[124]

Dorothy Denning, who was on the steering committee for the conference, recalled that the "idea was to bring together government and law enforcement with cyber people."[125] The conference succeeded admirably in this regard. Peter Neumann, as editor of the RISKS digest, reported that the meeting

> had a broadly based interdisciplinary audience, including law enforcers, lawyers, developers, vendors, marketers, computer scientists, (nonpejorative-sense) hackers, as well as crackers, whackers, and snackers (pejorative-sense hackers), trackers, backers, flackers (journalists), claquers, EFF-ers Kapor and Barlow, and a video crew one of whom was fresh from the Academy Awards Monday evening. Very few quackers (who duck the hard issues) or slackers.[126]

As Terry Winograd summarized, attendees "ranged from the sandals of Silicon Valley to the dark suits of Washington." In a widely reported keynote, Harvard Law Professor Lawrence Tribe called for a constitutional amendment to ensure that free speech rights would extend to cyberspace.

By the end of the meeting, many people agreed that the conference should be continued. Lance Hoffman, professor at George Washington University, reported: "Jim Warren and others twisted my arm with the result that I have become the general chairman of CFP-2, which will take place in Spring 1992 in Washington."[127] The conference continued through the 1990s with the support of ACM and several of its SIGs, including SIGCAS, SIGCOMM, SIGSAC, and SIGSOFT.

---

123. Ibid.

124. See First Conference on Computers, Freedom, and Privacy. Announcement posted in the RISKS forum. January 10, 1991. Available at http://catless.ncl.ac.uk/Risks/10.77.html#subj10.1

125. Denning oral history, 51.

126. P. G. Neumann. March 29, 1991. More on Computers, Freedom, and Privacy. Available at http://catless.ncl.ac.uk/Risks/11.35.html#subj3.1.

127. L. Hoffman. April 5, 1991. Computers, Freedom, and Privacy trip report. Risks Forum. Available at http://catless.ncl.ac.uk/Risks/11.40.html#subj1.1.

12.4 Conclusion

Between the late 1960s and mid-1990s, ACM's role in public debate about computer security and privacy changed dramatically. In the late 1960s and mid-1970s, ACM leaders avoided the association's official involvement in public debates over computer-related privacy. Instead, in their publications and meetings, ACM special interest groups discussed computer security and privacy as part of largely separate agendas. However, after the rise of computer networking prompted increased public debate about computer security and privacy, ACM became more involved in the public policy debate, and established ways of framing security were called into question. ACM institutionalized its involvement in public policy debates by establishing forums such as the Computers, Freedom, and Privacy (CFP) conference.

The CFP was only the beginning of ACM meetings focused on computer security. In 1993, ACM and the IEEE Computer Society established a joint Conference on Computer & Communications Security. SIGSAC, which had struggled to find a purpose in the early 1980s, grew increasingly active. In 1991, it began sponsoring a workshop at the Annual Computer Security Applications Conference, and in 1992 SIGSAC sponsored the first New Security Paradigms Workshop.[128] In 1995, SIGSAC also began sponsoring the conference on Role-Based Access Control (RBAC), which became the Symposium on Access Control Models and Technologies (SACMAT) in 2000.

ACM also became increasingly involved in public policymaking on computer security in the 1990s. In the early 1990s, ACM established the U.S. Public Policy Council, and its first major study examined the U.S. government's proposal to require that telephones and other electronics communications devices sold in the U.S. use the "clipper chip"—an encryption device which would keep the key in escrow with the government, so that agencies which gained the proper legal authority could eavesdrop on electronic communications.[129] *Commununications of the ACM* published considerable debate on the Clipper Chip, and the public policy council ultimately issued a statement against it. Barbara Simons, as chair of the council, testified before Congress against the Clipper Chip, and supported proposals to end export controls on encryption technologies.[130] ACM, along with many other opponents of the Clipper chip proposal, eventually succeeded in defeating it.

128. D. Faigin. 1992. Letter from the chair. *SIGSAC Rev.*, 10(4): 1–2.

129. B. Simons. February 10, 1994. Campaign and petition against Clipper. Available at http://catless.ncl.ac.uk/Risks/15.50.html#subj1.1.

130. Senate Committee on Commerce, Science, and Transportation. S. 1726, Promotion of Commerce Online in the Digital Era Act of 1996, or 'Pro-CODE' Act," June 26, 1996.

Although the widespread adoption of computer networking shaped ACM's leaders' conceptions of privacy and security, it by no means determined those understandings. I have argued that the SIGs encouraged distinctive ways of framing security and privacy in the 1970s, but that those framings were called into question in the 1980s. Was the government's monopoly on strong encryption and associated risks of surveillance greater than the risks to national security that came with the widespread availability of encryption? Were legislative or educational approaches to resolving the hacker problem a distraction from the need to implement better technical and managerial solutions? Who should be held accountable for vulnerable computer systems? Most of these questions are still with us today, as are ACM-sponsored forums for discussing them.

13

Hypertext, Digital Libraries, and Beyond: A History of ACM SIGWEB

Inna Kouper

13.1 Introduction

Professional organizations advance particular professions. In academia they also promote research and scholarship in relevant academic disciplines. The activities of academic professional organizations typically include organizing conferences and meetings, facilitating publications, supporting early career researchers and underrepresented groups, and rewarding excellence in research and education. Embedded in complex networks of actors and interests, organizations establish and negotiate practices, norms, and relationships that contribute to the development of research concepts and domains.

Research domains have their own trajectories. They develop within the elaborate environments of knowledge production as well as the structures of funding, disciplinary division, authority, and communication. Many research concepts have intricate histories of topical and terminological transformations; conceptual change is embedded in the creative practices that iterate within cognitive, cultural, and organizational dimensions.[1] While some dimensions of the history of scientific domains and concepts have been explored extensively,[2] the organizational aspect has received less attention.

This chapter brings together the notions of individual creativity, organizational history, and conceptual change in the examination of conceptual developments within a history of a particular organization. The chapter focuses on the Association for Computing Machinery (ACM) and its Special Interest Group on Hypertext and the Web (SIGWEB), and examines how hypertext and other concepts have been envisioned and practiced by those who considered their work relevant to this group. While many hypertext histories have been published, little has been written on the role of ACM in supporting the development of research domains and concepts.

The choice of hypertext and SIGWEB is not arbitrary. At once an area of innovative computing and a non-sequential form of knowledge, hypertext invited technological innovation, literary criticism, and multimedia experimentation.[3] The term existed before SIGWEB, but SIGWEB provided a venue for hypertext researchers and practitioners to exchange ideas and make this concept part of their professional identity. Both hypertext research and SIGWEB faced challenges from the Web. SIGWEB, initially named SIGLINK, had to reconsider its identity and embrace other concepts. SIGWEB and the conceptual boundaries it maintained and expanded provide a case to explore the relationship between professional organizations and the shaping of research areas, communities, and concepts.

This effort weaves together interrelated concepts and their meanings in the context of ACM SIGWEB through a series of quantitative and qualitative studies. The goal is to write a definitive history of SIGWEB and investigate the knowledge work of professional associations and the formation and maintenance of disciplinary and organizational fields. As a first step, a qualitative study was conducted in 2012–2014 that collected interviews, published documents, and archival materials and used discourse analysis to identify common themes in references to events and termi-

1. I. Hacking. 1999. *The Social Construction of What?* Harvard University Press, Cambridge, MA; T. J. Misa. 2015. Charles Babbage, Ada Lovelace, and the Bernoulli numbers. In R. Hammerman and A. L. Russell, editors. *Ada's Legacy: Cultures of Computing from the Victorian to the Digital Age*. Association for Computing Machinery and Morgan & Claypool, New York; J. J. Nersessian. 2010. *Creating Scientific Concepts.* MIT Press, Cambridge, MA.

2. U. Feest and F. Steinle, editors. 2012. *Scientific Concepts and Investigative Practice*. Walter de Gruyter, Berlin; M. Young and J. Muller. 2014. *Knowledge, Expertise and the Professions*. Routledge, London/New York.

3. G. P. Landow. 2006. *Hypertext 3.0: Critical Theory and New Media in an Era of Globalization*. Johns Hopkins University Press, Baltimore.

nology across the documents.[4] The interviews included eight interviews with ACM SIGWEB chairs and administrators conducted for this project and sixteen interviews conducted by SIGWEB.[5] The archival documents came from the ACM archives at the Charles Babbage Institute (CBI) at the University of Minnesota; the published documents were collected via extensive snowballing search of online databases and the web.

The chapter synthesizes and highlights the events around the creation and development of ACM SIGWEB and its connections to hypertext and other related concepts. It begins with a brief history of the ACM and uses this history to outline a model for describing an organization with an ever-expanding research domain. It then describes the early days of SIGWEB and its roots in hypertext. Next, the chapter traces the history of SIGWEB, including its name change, and reflects on the concepts that became relevant to the group over time. The chapter concludes by discussing a possible framework for studying the co-evolution of organizations and research fields.

13.2 The Association for Computing Machinery (ACM)

The history of ACM begins in the 1940s with a shift from analog to digital computing and a substantial increase in the number of machines that were built to perform calculations. In 1947 at the first Symposium on Large-Scale Digital Calculating Machinery at Harvard University the attendees endorsed a suggestion put forward by Samuel H. Caldwell to create an informal organization for free communications in the new field of computing machinery.[6] Caldwell called for cooperation between commercial, academic, and government researchers and for openness that was lacking in military-funded research.[7] The suggestion resulted in the creation of the *Eastern Association for Computing Machinery.*[8]

The formation of the ACM was supported enthusiastically by some and met with opposition and skepticism by others. Edmund Berkeley was an active proponent of the association, which fit his vision of the promises of computing applied to human thinking and decision-making.[9] He drafted the proposal of the new organization

4. M. Wetherell, S. Taylor, and S. J. Yates. 2001. *Discourse as Data: A Guide for Analysis*. Sage Publications, London.

5. Available at http://www.sigweb.org/community/interviews.

6. S. B. Williams. 1954. The Association for Computing Machinery. *J. ACM*, 1(1): 1–3.

7. A. Akera. 2007. Edmund Berkeley and the origins of ACM. *Commun. ACM*, 50(5): 30–35.

8. F. L. Alt. 1962. Fifteen years of ACM. *Commun. ACM*, 5(6): 300–307.

9. Akera, Edmund Berkeley and the origins of ACM.

and was instrumental in recruiting members throughout the U.S.[10] At the same time, such established figures in computing as John von Neumann, George Stibitz, and Samuel Williams were emphatically against it. Beyer also describes how IBM developers resisted the selection of the executive council during the first meeting at Columbia University, seeing an attempt to create a "self-adulation society for the benefit of non-IBM people."[11] Nevertheless, Berkeley along with John Curtiss and others, continued in their efforts to build the ACM, helping it gain legitimacy and broaden participation. Close interactions with leading figures who did not join the ACM, e.g., Howard Aiken, John Mauchly, and John von Neumann, ensured the meetings' cutting-edge character and breadth.

The ACM in its first several years was driven by the desire for open communication, rapid exchange of ideas, and collaboration between industry, government, and academic research. Throughout the 1950s–1960s it strengthened its academic focus, which came as a mixed blessing. Some industry figures viewed ACM and its leadership as "guys with their heads in the cloud worrying about Tchebysheff polynomials."[12]

In the 1960s–1970s the association actively discussed the need to expand its interests and be inclusive of the business communities, with various attempts to develop academic standards and to take part in professional certification. Despite its academic orientation, the breadth of ACM allowed it to grow and accommodate the changing technologies and interests. Identifying like-minded individuals and establishing a common vocabulary were foundational for consolidating researchers and practitioners and establishing computer science as a professional field.[13]

The early conferences focused on computing and math, with a particular emphasis on hardware, numerical analysis, and logical design; in other words, the organization's conceptual focus was machinery.[14] In the 1970s the focus expanded to computing techniques and appropriate languages for information processing of

10. B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals*. Association for Computing Machinery and Morgan & Claypool, New York.

11. K. W. Beyer. 2009. *Grace Hopper and the Invention of the Information Age*, p. 167. MIT Press, Cambridge, MA.

12. N. L. Ensmenger. 2012. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*, p. 172. MIT Press, Cambridge, MA.

13. B. Longo. 2007. ACM established to develop communication about computing. *Commun. ACM*, 50(5): 27–29.

14. American Mathematical Society. 1948. Mathematical tables and other aids to computation. *News* 3(22): 132–134. Available at http://www.jstor.org/stable/2002605; Williams, The Association for Computing Machinery.

data of all kinds and to simulation processes.[15] The establishment of SIGs, many of which were founded in the 1960s–1970s,[16] helped to sustain disciplinary diversity of the ACM. The present constitution describes ACM's purpose as "advancing the art, science, engineering, and application of information technology,"[17] thereby broadening the focus even more to the physical, logical, and social processes of information processing.

This brief history of ACM illustrates a successful path from a small group of active enthusiasts to the large international organization. Several factors contributed to ACM's success, including a shared interest in a problem area; willingness to share and cooperate; mechanisms to maintain group cohesiveness and increase influence; control over channels of formal and informal communication; and financial stability to expand its operations. The path also had many contingencies, including establishing academic legitimacy, professionalizing computer science, reconciling competing interests, and accommodating the broadening research and vocabularies.

Similar themes of cohesiveness, communication, finances, and conceptual changes were found in the SIGWEB communications and documents. As shown in this chapter, SIGWEB has some similarities to the history of ACM—it began as a small group of enthusiasts who shared an interest in hypertext and succeeded in creating a cohesive group with a substantial amount of influence. The highly academic status of SIGWEB's main conference *Hypertext* also proved to be a mixed blessing as the group almost overlooked the development of the Web and was rather late embracing other concepts. Unlike ACM, the SIG had difficulties maintaining group's cohesiveness and financial stability over time and accommodating changes in interests and technologies.

13.3 ACM SIGWEB

The importance of capturing organizational history is recognized by SIGs and ACM. In addition to the ACM archive at CBI, nineteen SIGs maintain history wikis or websites where they collect resources relevant to their history.[18] In addition to recording the events and contributors of the past, the histories highlight the paths

15. L. Revens. 1987. The first 25 years: ACM 1947–1962. *Commun. ACM*, 30(10): 860–865.

16. A. Cochran. 1987. ACM: The past 15 years, 1972–1987. *Commun. ACM*, 30(10): 866–872.

17. Available at http://www.acm.org/about/constitution.

18. See ACM History Committee: SIG History Activities. Available at http://history.acm.org/content.php?do=sighistory.

that groups took in defining their focus and activities. Thus, the group SIGMOBILE started as a group that focused on wireless and mobile technologies and expanded into "an organization that fosters research in the field of mobility and tetherless ubiquitous connectivity."[19] Akin to the wider ACM, this group's scope and self-definition expanded from specific technologies into broader concepts.

Efforts to document SIGWEB history have so far been modest. In 2013 a short article in the SIGWEB *Newsletter* briefly described SIGWEB activities and urged the membership to join the effort of other SIGs to contribute to history wikis.[20] The ACM archive contains older chapter activities reports; SIGWEB is also mentioned in many annual ACM meeting reports.[21] This chapter is the first attempt to document SIGWEB history systematically.

SIGWEB started as a group that focused on hypertext and hypermedia. Hypertext is a text that offers mechanisms for non-sequential reading; it is a representation of materials on the computer that links them together and provides multiple navigational paths.[22] Many histories of hypertext describe the origins of the term and emphasize the computing component of it.[23] Some histories go beyond computing and incorporate a broader range of precursors, especially from the European traditions of literary theory, aesthetics, and information science.[24]

The computing history of hypertext begins with the director of the U.S. Office of Scientific Research and Development Vannevar Bush and his 1945 article "As

19. Available at http://www.sigmobile.org/about/history.html.

20. B. White. 2013. SIGWEB History Project. *ACM SIGWEB Newslett.*, (Spring 2013): 1–5.

21. ACM. 1967–1989. Published annual reports with summary of activities, membership data and financial statement (Box 1, Folders 15–16). Association for Computing Machinery Records, 1947–2009. University of Minnesota Libraries, Charles Babbage Institute. Minneapolis, MN; ACM. 1976, 1981, 1991. Annual meetings (Box 4, Folders 5–7); ACM. 1985, 1987–1989. Annual reports (Box 2, Folders 4–8); ACM. 1991–2003. Annual reports (Box 2, Folders 9–21); ACM. 1988–2002. SIG chapters activities reports (Box 39).

22. T. H. Nelson. 1965. Complex information processing: A file structure for the complex, the changing and the indeterminate. In *Proceedings of the 20th National Conference*, pp. 84–100. ACM Press, New York; J. Nielsen. 1995. *Multimedia and Hypertext: The Internet and Beyond*. Morgan Kaufmann, San Francisco.

23. B. Barnet. 2013. *Memory Machines: The Evolution of Hypertext*. Anthem Press, London; J. Conklin. 1987. Hypertext: An introduction and survey. *IEEE Comput.*, 20(9): 17–41; D. H. Jonassen. 1989. *Hypertext/Hypermedia*. Educational Technology Publications, Englewood Cliffs, NJ.

24. E. J. Aarseth. 1997. *Cybertext: Perspectives on Ergodic Literature*. Johns Hopkins University Press, Baltimore; J. D. Bolter. 2001. *Writing Space: Computers, Hypertext, and the Remediation of Print*. Lawrence Erlbaum Associates, Mahwah, NJ; A. Kitzmann. 2006. *Hypertext Handbook: The Straight Story*. Peter Lang, New York; A. Wright. May 2014. The secret history of hypertext. *The*

We May Think,"[25] where he described the great challenge for the post-World War II times—to organize the ever-growing volumes of information. Bush proposed a mechanized device, *memex*, to supplement human memory and to store books and communications and retrieve them with speed and flexibility. The speed and flexibility were supposed to come from the use of then-modern technologies, such as translucent screens, microfilm, and dry photography, and from associative indexing that would create "trails" among documents via coded words.

Bush's idea of using technology to enhance human memory and information retrieval inspired and influenced many computer scientists. His vision of associative interconnected knowledge supported by computer technology was a source of inspiration for Douglas Engelbart and Theodore Nelson, two other visionaries who had an immense impact on hypertext developments.

Engelbart's inspiration and views on hypertext as a system that could improve knowledge work and communication came from at least two sources—Bush's ideas of associative linking and Benjamin Lee Whorf's ideas of linguistic connections.[26] While associations have a rather accidental character and may reflect the thinking processes of an individual, connections reflect the experiences of social groups and can be used to improve communication. Connections are more controlled relationships between concepts and their use in particular languages. Engelbart's work focused on enabling controlled connections rather than on supporting accidental associations. His system NLS (oN-Line System), designed and implemented in the 1960s, enabled collaborative work and linking between documents.[27] In addition to linking, it incorporated many innovations that are now a standard part of collaborative work, including multiple windows, document version control, shared-screen teleconferencing, and context-sensitive help.

Nelson's vision of hypertext also included collaboration and collective work, but his vision had two other major themes: (1) the world of literature (taken broadly as

Atlantic. Available at www.theatlantic.com/technology/archive/2014/05/in-search-of-the-proto-memex/371385/.

25. V. Bush. July 1945. As we may think. *The Atlantic*, pp. 101–108. Available at www.theatlantic.com/magazine/archive/1945/07/as-we-may-think/303881/

26. T. Bardini. 2000. *Bootstrapping: Douglas Engelbart, Coevolution, and the Origins of Personal Computing.* Stanford University Press, Stanford; T. Bardini. 2006. Bridging the gulfs: From hypertext to cyberspace. *J. Comput.-Mediat. Commun.* 3(2).

27. D. C. Engelbart. 1986. The augmented knowledge workshop. In *Proceedings of the ACM Conference on the History of Personal Workstations*, pp. 73–83. ACM Press, New York; D. C. Engelbart. 1995. Toward augmenting the human intellect and boosting our collective IQ. *Commun. ACM*, 38(8): 30–32.

information) is a system of interconnected works that are impossible to separate into subjects and domains;[28] and (2) computers are supposed to bring freedom and creativity rather than exert control. Interactivity and interconnectedness, according to Nelson, were intrinsic to the computer medium, hence the departure from paper that cannot represent the complexity of textual links and media performances. Despite some parallels between his ideas and the Enlightenment's encyclopedic efforts of cross-referencing, keyword navigation, and subject mapping,[29] hypertext systems were to enable innovative non-sequential presentations of textual, graphical, and other material and offer "total freedom from arbitrary categorizing and chopping."[30]

A joint story of hypertext and SIGLINK/SIGWEB begins in 1987, when ACM sponsored the *Hypertext '87* workshop, held at Chapel Hill, North Carolina. At that time hypertext was an area that explored system design, interfaces, and applications of hypertext systems with multiple papers presented at the ACM conferences, such as the conference on document processing systems DOCPROCS or the conference on management of data SIGMOD. One of the most important concepts in the earlier history of hypertext—the link—gave the group its first name, SIGLINK.

Hypertext'87, organized by John B. Smith from University of North Carolina and Frank Halasz from Microelectronics and Computing Technology Corporation was "the first large workshop/conference devoted to the subject."[31] It featured many commercial and academic hypertext systems and design approaches of those times, including NLS, KMS, Xanadu, HyperCard, Document Examiner, Storyspace and many others.[32] The workshop discussed many current issues and challenges and identified an agenda for the future, including navigation via search and query, multiple views on content, versioning, and computational access to information.[33]

28. T. Nelson. 1974. *Computer Lib/Dream Machines*. Mindful Press, Sausalito, CA.

29. D. Rosenberg. 2015. History debugged. In D. R. Dechow and D. C. Struppa, editors. *Intertwingled: The Work and Influence of Ted Nelson*, pp. 91–103. Springer, Heidelberg.

30. Nelson, *Computer Lib/Dream Machines*, 328.

31. NSF. 1987. Award #8715704–Hypertext 87: Workshop on Systems, Applications, and Issues. Available at http://www.nsf.gov/awardsearch/showAward?AWD_ID=8715704&HistoricalAwards=false.

32. J. B. Smith and F. G. Halasz. 1987. *Proceedings of the ACM Conference on Hypertext*, ACM Press, New York.

33. F. G. Halasz. 1987. Reflections on NoteCards: Seven issues for the next generation of hypermedia systems. In *Proceeding of the ACM Conference on Hypertext–HYPERTEXT '87*, pp. 345–365. ACM Press, New York.

The *Hypertext'87* keynote given by Andries van Dam highlighted the positive energy of the workshop:

> Well, here we are at the first hypertext workshop and we have to ask, perhaps rhetorically, has hypertext arrived? has the millennium arrived? . . . When you look at the publicity and at systems such as Xerox's NoteCards, Owl's Guide, and especially Apple's HyperCard, you get the impression that we are about to go over the knee of the exponential curve. I really believe we are. . . . it will come through the HyperCard phenomenon, through the proceedings of this workshop, and the networking that this workshop will engender. So my summary is that no, the millennium has not arrived yet, but we are about to go over the knee.[34]

At *Hypertext'87*, Robert Akscyn, a computer scientist and president of Knowledge Systems, Inc., John J. Leggett from Texas A&M University, and Janet Walker from Symbolics, Inc. discussed the creation of a working group that could focus on the development of common terminology and subsequently a common model of hypertext systems. The discussion was followed by a workshop organized by Leggett and Walker in October 1988 at the Dexter Inn in Sunapee, New Hampshire. The name of the hotel gave name to the group and subsequently to the Dexter Hypertext Reference Model, a three-layer model that conceptualized the node/link structure of early hypertexts and outlined more complex features of the hypertext systems, such as multidirectional links and aggregations of single nodes into composite structures.[35]

The Dexter group consisted of 13 "carefully selected hypermedia designers" who played a significant role in hypertext and hypermedia and understood the rationale behind such systems as well as their technical details.[36] It was a tight community of researchers and designers who were explicitly interested in hypertext and often crossed the boundary between academic research and commercial system development. The Dexter group represented hypertext/hypermedia systems such as KMS, NoteCards, HyperCard, HyperTies, and HAM, which were developed within or in collaboration with commercial companies.

Many of the members of the Dexter group became active participants in the Hypertext (HT) conference and other SIGWEB activities. "Simply by happenstance,"

---

34. A. van Dam. 1988. Hypertext '87 keynote address. *Commun. ACM*, 31(7): 887–895. Available at http://cs.brown.edu/memex/HT_87_Keynote_Address.html.

35. F. G. Halasz and M. Schwartz. 1994. The Dexter hypertext reference model. *Commun. ACM*, 37(2): 30–39.

36. K. Grønbæk and R. H. Trigg. 1999. *From Web to Workplace: Designing Open Hypermedia Systems*. MIT Press, Cambridge, MA.

Figure 13.1 **Hypertext was publicized at ACM SIGGRAPH (1988).** (Image courtesy of the Charles Babbage Institute Archives, University of Minnesota Libraries)

as Robert Akscyn put it, he became one of the leaders of that group and worked along with others to organize the next HT conference in 1989 and to propose a special interest group on hypertext to the ACM—SIGLINK. Similar to the ACM's beginnings, some members of that community were not convinced that the SIG would be a success.

SIGLINK was one of the two groups approved out of twelve requests in November 1990.[37] The group's purpose was to advance the interests of computing profes-

37. ACM, Annual reports (Box 2, Folders 9–21).

sionals engaged in the use of hypertext and other hypermedia to store and retrieve computer-based information. In 1991 SIGLINK consisted of 217 members; in 1992 it grew to 989 members. Several people were involved in the organization of the first few HT conferences prior to the SIG, including (already mentioned) John Smith and Frank Halasz, and it was reasonable that they joined the conference activities under the SIG umbrella. Robert Akscyn was instrumental in persuading the European hypertext community to alternate conferences under SIGWEB sponsorship in the U.S. in odd years and in Europe in even years.

At the beginning much of the hypertext research focused on the design and architecture of hypertext systems and the mechanisms to support links and non-sequential document structures. The early ACM HT conference proceedings described many stand-alone systems and models and proposed innovations that supported cognitive processes, collaboration, extensive text annotation, and navigation through large numbers of documents. Most of the approaches to interaction focused on human-computer interaction, rather than on user-to-user interaction or sharing of information.

In addition to the strong emphasis on systems, navigation, and representation of linked structures, the early hypertext community of the 1980s–1990s was also concerned with social and cultural issues and explored the notions of creativity, writing, and humanities scholarship. J. D. Bolter and M. Joyce presented their system for authoring hypertext fiction at *Hypertext'87*.[38] Writers and humanities researchers such as Mark Bernstein and Elli Mylonas were actively involved in the early conferences and in the SIGLINK/SIGWEB activities. In 1989, for example, the conference had a panel on hypertext, narrative, and consciousness that explored the interconnections between information and literary theories and its implications for texts and systems.

The diversity of SIGLINK and its humanities–sciences–industry connections were seen as a strength of the group, and the program committees worked hard to maintain that diversity for some time and to keep topics wide and open. The hypertext technology was discussed in the educational and social contexts as having an impact on learning and information processing. Legal issues and ownership and copyright were also a large part of the discussion.[39] The dialogue between

38. J. D. Bolter and M. Joyce. 1987. Hypertext and creative writing. In *Proceedings of the ACM Conference on Hypertext–HYPERTEXT '87*, pp. 41–50. ACM Press, New York.

39. J. B. Smith and S. F. Weiss. 1988. Hypertext. *Commun. ACM*, 31(7): 816–819.

scientists and humanists was considered very important for the future of hypertext technology.

The interdisciplinarity and excitement of the first few conferences generated a sense of camaraderie in the community that strived to do research and creative experimentation in a relatively new area. A core group contributed to running the conferences and ensuring its high academic quality. The community remained tightly knit and continued to focus on systems design. The richness of the technical challenges of linking and navigation was partly the reason that the community did not look beyond the immediate research domain into the broader areas of user needs and applications. As a result, the direction in which SIGLINK was going in terms of conferences overlooked the importance of the gradual shift towards networked distributed systems and ignored the early developments of the World Wide Web.

The early developments of the WWW and the work of Tim Berners-Lee focused on sharing information and documents among many people. His proposal for a distributed hypertext system acknowledged the richness of research in this area, but it insisted on several practical principles and constraints, such as simple access and navigation, openness and connectivity, use of the existing data, and implementation of hypertext views rather than hypertext databases.[40] In his original proposal, Berners-Lee clearly stated his preference for simple linking over interface and architecture complexity: "We should work toward a universal linked information system, in which generality and portability are more important than fancy graphics techniques and complex extra facilities."

Berners-Lee submitted a paper that described his approach to the HT conference in 1991. It was *rejected* as being weak on research and not addressing the problems of broken links properly; it also did not contain references to the relevant work in the field.[41] The practical simplicity of Berners-Lee's ideas ignored the challenges of bi-directional links and multi-way navigation, preferring to focus on networking and scale. The sophisticated creativity that hypertext systems promised in humanities research was also ignored in the limited document structuring proposed by HTML. The rejected paper was accepted as a demo presentation.[42]

---

40. T. Berners-Lee. 1990. The original proposal for the WWW. Available at http://www.w3.org/History/1989/proposal.html (accessed December 19, 2015).

41. T. Berners-Lee and M. Fischetti. 1999. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. Harper, San Francisco.

42. R. Cailliau. 1995a. A short history of the Web. Available at http://www.netvalley.com/archives/mirrors/robert_cailliau_speech.htm (accessed January 21, 2016).

In the mid-1990s the vision of hypertext research still focused on standalone systems that explored architectural, aesthetic, and design complexities. The first Engelbart Best Paper Award at *Hypertext'96* was given to the paper "HyperCafe: Narrative and Aesthetic Properties of Hypervideo."[43] The paper discussed a hypermedia prototype that allowed the user to navigate through a "virtual café" and listen to video clips that represented multiple ongoing conversations by selecting certain points in time, i.e., narrators or parts of the conversation, and in space, i.e., tables or areas of conversations. As the 1996 SIGWEB annual report stated,[44] there was a consensus that this paper represented the interests of SIG members, reflected the state of hypertext scholarship, and contributed to the field's vision of the future. The future, thus, was in labor-intensive, manually implemented experiments that produced sophisticated aesthetic experiences for individual users.

Despite evidence that in the 1990s the Web and hypertext research existed in somewhat parallel universes, there was overlap. SIGLINK members attended the World Wide Web conferences that started in 1993, the HT and WWW conferences held joint sessions, and Berners-Lee was a keynote speaker at the European Conference on Hypertext ECHT'94. Gradually, the concept of the Web became part of the HT conferences. In 1996 two papers appeared in its proceedings: "A study of navigational support provided by two World Wide Web browsing applications" and "Browsing the WWW by interacting with a textual virtual environment—a framework for experimenting with navigational metaphors." The SIGLINK/SIGWEB newsletter started a new column on web research in December 1995.

Rather than pursuing further connections to the web research communities, SIGLINK expanded its reach into other domains, particularly, the domain of digital libraries. In 1994 Richard Furuta and John Legget, both active members of SIGLINK, chaired the first Annual Conference on the Theory and Practice of Digital Libraries (DL'94). The conference was not yet an ACM conference,[45] but it was organized in cooperation with ACM SIGLINK and three other SIGs. Robert Akscyn,

43. N. Sawhney, D. Balcom, and I. Smith. 1996. HyperCafe: Narrative and aesthetic properties of hypervideo. In *Proceedings of the Hypertext '96 conference*, ACM Press, Washington, DC.

44. ACM, SIG chapters activities reports (Box 39).

45. In 1994 and 1995 DL conferences were organized and sponsored by academic institutions in cooperation with ACM SIGs, in 1995–2000 there were two parallel DL conferences, organized by ACM and IEEE. In 2000 these two conferences merged, and since 2001 it has been a joint ACM/IEEE conference on digital libraries (JCDL).

Edward A. Fox, and Catherine C. Marshall were members of the organizing committee for DL'94.[46] Akscyn and his company Knowledge Systems, Inc. provided support for awards at both HT and DL conferences ($1,000 each for the Douglas Engelbart and the Vannevar Bush awards).

The domains of hypertext and digital libraries had overlapping research trajectories in the 1990s, which was one of the reasons for the same researchers contributing to both, meeting at the same venues, and developing closer relationships. Similar to hypertext, digital libraries research traced its origins back to Bush and the "memex" and considered the use of innovative technologies for knowledge management and discovery as one of its primary goals.[47] Both domains explored document structuring and representation, browsing and retrieval, and multimedia; they both addressed the challenges of distributed storage and networked access and attracted researchers interested in interdisciplinary collaborations and support of multiple applications in science, literature, history, and education.

One of the key differences between hypertext and digital libraries was that the latter was concerned with extending existing library services, while hypertext research strived to transcend existing modes of reading and writing. Nevertheless, there were many researchers who were successful in straddling the two fields and actively pursuing both. Catherine Marshall, for example, won both the HT Engelbart and DL Bush awards for her papers "Toward an ecology of hypertext annotation" and "Making metadata: a study of metadata creation for a mixed physical-digital collection," an example of how social sciences ethnographic approaches were applied in the studies of hypertext and digital library communities.

In the second half of the 1990s the vocabulary and technologies continued to change. The research in digital systems shifted toward large-scale collections, distributed storage, and retrieval, networking, and collaborative access. The WWW was conquering the world and the divide between hypertext and the Web led to more misunderstandings than opportunities for research. As the Web's underlying technology was essentially hypertext, the differences between the use of this word in the web's HTML, HTTP, and hyperlinking and in the rest of hypertext were becoming harder to explain. Some researchers perceived non-Web hypertext as too restrictive. With the relatively low visibility of SIGLINK and the increasing visibility

46. Available at http://www.csdl.tamu.edu/DL94/committees.html.

47. E. A. Fox, R. M. Akscyn, R. K. Furuta, and J. J. Leggett. 1995. Digital libraries. *Commun. ACM*, 38(4): 22–28; L. Candela, D. Castelli, and P. Pagano. 2011. History, evolution, and impact of digital libraries. In I. Iglezakis, T. Synodinou, and S. Kapidakis, editors. *E-Publishing and Digital Libraries: Legal and Organizational Issues*, pp. 1–30. Information Science Reference, Hershey, PA.

of the Web, the new generations of researchers did not see SIGLINK venues as their first choice. To attract broader audiences, SIGLINK executive committee pursued two avenues: (1) change the name of SIGLINK and open it to web studies and (2) engage with the International World Wide Web conference (WWW).

The proposal to change the name was circulated among the SIGLINK membership. Not everyone agreed that changing the name was the right way to go; some insisted on the difference between hypertext-like interfaces to access resources in a distributed system (i.e., WWW) and the genuine hypertext structures with deeper research potential that existed outside of the Web.[48] Nevertheless, after a number of private and public discussions, it was decided that "LINK" did not represent the domain well. A formal proposal was presented at the ACM SIG Board Meeting in 1998 with the following rationale.[49]

> In the early 1990s, SIGLINK was too new a SIG and missed the opportunity to take advantage of the birth and the extraordinarily fast growth of the WWW. It did not explicitly host the emerging domain. . . . These two areas [SIGLINK and WWW] are coming closer together, since, in fact, they share the same underlying domain, rely on the same fundamentals, and often share the same individuals. . . . To conclude, the term SIGWEB is representative of our focus. For our membership, a 'WEB' is a general concept of the domain, which includes the WWW. For newcomers, it also welcomes work on the WWW.

SIGLINK was renamed SIGWEB. Despite some concerns, it did not mean that the interests of the community had shifted completely to the World Wide Web. As the proposal stated, "web" was a more general concept that denoted networks and relationships in information. The community was broader than WWW, and the hope was that a new name would open doors to web-enabled research construed broadly.

In the late 1990s–early 2000s both SIGWEB as an organization and hypertext as a research domain experienced challenges. Hypertext conference attendance as well as SIGWEB membership was shrinking. While in 1991 the conference attendance was reported to be around 450 attendees, in 2000 and 2001 it was 130 and 170, respectively.[50] The core hypertext community continued to be engaged in the conference and other activities, but the broader audiences that were hoped for

48. B. White. 2013. SIGWEB History Project. *ACM SIGWEB Newslett.*, (Spring 2013): 1–5; see also a discussion in the ACM SIGWEB *Newslett.*, 7, 1998).

49. ACM, SIG chapters activities reports (Box 39).

50. ACM, Annual reports (BOX 2, Folders 9–21; ACM, SIG chapters activities reports (Box 39).

with the rebranding of the SIG did not materialize. There are multiple reasons for the shrinking of SIG memberships, common to many SIGs at the time.[51] But in the case of SIGWEB there were particular challenges in bridging the divide between the technical, applied, and the creative strands within the domain.

The challenges of supporting interdisciplinary contributions within the SIGWEB hypertext community had to be addressed in conference programming, in vocabulary convergences, in the perceived benefits of interdisciplinarity to individual careers and the field, and in the mere willingness of the diverse group to spend their time and resources together. To some computer scientists the conference and the group was not technical enough, emphasizing more of the information science issues. To others, the literary and interactive scholarship moved into other venues that better suited their interests and careers. On top of that, the literary hypertext and scholarship had not grown as expected; and researchers in creative experimentation, while somewhat expanding into other mediated environments, such as gaming, virtual worlds, and the Internet, remained wary of a deep dive into technical proficiency.[52]

Lack of easily available non-WWW examples of hypertext systems slowed the progress in non-literary fields as well. Within SIGWEB, efforts were made to make hypertext useful to SIG members and broader audiences. In 1988 the ACM released the first hypertext product *Hypertext on Hypertext,* which contained the major articles from the special issue on hypertext of the *Communications of the ACM* (1988). The product was released in three versions, for IBM PC (HyperTIES version), for Apple Macintosh (HyperCard version), and for Sun workstations (KMS version).[53]

It was a successful small-scale hypertext publishing project that sold approximately 2,200 copies and recovered the direct expenses of its production.[54] At the same time, it was a hand-crafted collection, and the lessons from its creation were the difficulties of creating an elegant and efficient product and in subjectivity of

51. ACM. 1995. Annual Report of the SIG Board for the period July 1, 1994–June 20, 1995. Available at http://oldwww.acm.org/sig_board/fy95annrpt/report.html.

52. M. Bernstein. 1999. Everything is intertwingled: Asking again. In *Proceedings of the ACM Conference on Hypertext–Hypertext'99*, ACM Press. Available at www.eastgate.com/ht99/slides/Welcome.htm.

53. B. Rous, B. Shneiderman, N. Yankelovich, and E. Yoder. 1989. Lessons learned from the ACM Hypertext on Hypertext Project. In *Proceedings of the Second Annual ACM Conference on Hypertext*, (HYPERTEXT '89), pp. 385–386. ACM, New York.

54. ACM, Annual reports (Box 2, Folders 4–8).

links and associations.[55] Other ideas, such as LinkBASE, a database management system for information on events, publications, organizations, and people related to ACM SIGWEB[56] or an attempt to maintain a list of relevant course listings did not endure at all.

The challenges of maintaining a financially viable organization, while relying predominantly on the work of volunteers, led to some gaps in the management and activities of SIGWEB. It suffered through serious budget deficits and difficulties with conference scheduling and organization. Commercial and organizational sponsorship for SIGWEB activities was limited and focused primarily on sponsoring awards and technical panels and demos. As a result, the newsletter effort that had been an important informal channel of communication and a requirement for the SIGs stopped in 2000. In 2002 the status of SIGWEB was changed to transitional, and it became a conference-only SIG instead of a full-service SIG.

After several years of difficulties within the SIG, its leadership changed. The 2003 SIGWEB annual report began with the following:

> This has been a quiet year for SIGWeb. We continue to have several strong conferences, and have recently embarked on a new conference sponsorship (DocEng). We are very close to submitting all the necessary materials to our membership so we can come out of transition in September. We are also in the process of transitioning to a new leadership team.[57]

As SIGWEB gained new leadership, it continued to recognize the diversity of its membership and the need to expand the scope beyond hypertext and WWW. The role of technology in augmenting human intellect and creativity articulated by hypertext pioneers was seen to be realized in many forms, including the Web, digital libraries, and knowledge management systems. Another excerpt from the 2003 report reads like a forward-looking vision of broad research agenda, interdisciplinarity, and research-practice connection:

> From cognitive psychologists to ethnographers to anthropologists, SIGWEB embraces those researchers and practitioners that address how people use computers, so that better tools for augmenting the human intellect can be built. SIGWEB

55. E. Barrett. 1991. *The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information*. MIT Press, Cambridge, MA.

56. V. Balasubramanian, M. Bieber, and T. Isakowitz. 1996. *Systematic Hypermedia Design*. Available at http://archive.nyu.edu/bitstream/2451/14198/1/IS-96-09.pdf.

57. ACM, Annual reports (Box 2, Folders 9–21).

> also balances the findings of the research world with the experiences of the practical world, in which our ideas and theories are tested daily.

The early 2000s saw an expansion of SIGWEB activities into multiple domains. In addition to sponsoring two main conferences—Hypertext (HT) and the joint ACM/IEEE conference on digital libraries (JCDL)—SIGWEB engaged in supporting the Conference on Information and Knowledge Management (CIKM), Symposium on Document Engineering (DocEng), the Conference on Web Search and Data Mining (WSDOM), and, later, the Web Science Conference (WebSci). Additionally, various cooperation agreements have supported many other workshops, symposiums and conferences. Broadening conference support allowed SIGWEB to expand its reach, avoid unnecessary competition, and stabilize its finances.

At the same time the efforts of SIGWEB to become "the Web" SIG were not successful. First, attempts to incorporate the Web into the hypertext community had failed earlier in 1991 and 1993.[58] Then, a "major disconnect happened":

> The Web community created its own WWW conference series—there were two in 1994—the first at CERN in Geneva and the second at the home of Mosaic in Chicago. There was no ACM Hypertext conference in 1995 (there were two WWW conferences) the next in the series was HT'96 in Washington. Annoyingly, the WWW conference in 1997 (Santa Clara) clashed with HT'97 (Southampton). Hypertext researchers had to choose which conference to go to. . . . Many people at the first WWW conference thought the Web was the first "hypertext" system.[59]

The WWW conference organizers did not see ACM and SIG sponsorship as valuable since the conference quickly gained popularity and prestige on its own—the first WWW conference already had 380 participants.[60] It also developed its own model of running a conference with a special standing conference committee (IW3C2) and academic partners as organizers, W3C as a standing partner, and commercial entities as sponsors. At the same time, the academic quality of ACM conferences was of interest to the WWW conference; so eventually, ACM arranged

58. R. Cailliau. 1995b. How it really happened. Available at http:/www.netvalley.com/archives/mirrors/Robert_Cailliau_How_It_Really_Happened.htm (accessed January 21, 2016).

59. W. Hall. 2007. Back to the future with Hypertext: A tale of two or three conferences. In *Proceedings of the Eighteenth Conference on Hypertext and Hypermedia*, (HT '07), pp. 179–180. ACM, New York.

60. R. Cailliau. 1994. First International Conference on the World-Wide Web. Available at http://www94.web.cern.ch/WWW94/ (accessed January 21, 2016).

publication of the WWW proceedings in its digital library and SIGWEB appeared as a conference sponsor in 2010.

Another opportunity to bring hypertext and the Web closer came in 2010 when the WebSci conference ran as a subsidiary of the WWW conference in Raleigh, North Carolina. Web science is essentially the study of the Web, which encompasses the semantic web and examines the mutual impacts of the Web and society.[61] Through the facilitation of researchers, such as Wendy Hall who had strong ties to ACM, SIGWEB, and the world of WWW, the SIGWEB executive committee met with the WebSci steering committee and Ethan Munson, the chair of SIGWEB, proposed for WebSci to become an ACM conference. The agreement was reached that ACM would sponsor the conference for three years and after that unless either side rejects it, it would become a permanent meeting run by SIGWEB. In 2013, WebSci became a SIGWEB conference.

In the 2010s SIGWEB broadened its research focus to linked information in all forms and technologies.[62] It specified the Web as one of the famously realized hypermedia systems, but also included other network- and link-based approaches to information and knowledge, such as social networks, semantic web, and search and recommendation engines. The membership increased to about 500, and the activities expanded from conferences and awards to supporting student travel and providing web services, such as a dissertation repository.[63] Even though the word "hypertext" in the group's description may seem somewhat old-fashioned, the underlying idea of fast and intelligent linking across vast amounts of information is what SIGLINK/SIGWEB has been working on since the beginning. Now that linking is re-conceptualized as a main mode of interaction, the field of hypertext research can embrace applications (social media, repositories), computation (web and data science), knowledge inferences (the semantic web), and media design (mobile technologies and non-textual inputs).

13.4 Conclusion: Toward a Model of Epistemic Work in Professional Organizations

This chapter is the first systematic history of ACM SIGWEB. It describes how the group came into being via organizing around the concept of hypertext and how the group's focus changed over time due to the changing interests of its audiences

61. Hall, Back to the future with hypertext: A tale of two or three conferences.

62. Available at http://sigweb.org/about-sigweb/mission.

63. Available at http://sigweb.org/community/thesis.

and members and the changes in technology and its applications. The chapter presents a trajectory of SIGWEB as a specialized group that evolved to understand the importance of topical heterogeneity and to accept a diverse and changing research base.

As a professional interest group, SIGLINK/SIGWEB played an important role in gathering the core group of hypertext researchers in the 1980s and providing a venue for them to network and exchange ideas. At the beginning, SIGLINK's community was smaller and the like-minded individuals interacted informally on a regular basis. Shared interests, strong relationships, and the involvement of prominent researchers enabled the group's success. As Robert Akscyn emphasized in his interview for this study, "people and their conversations in the hallway, over drinks, over meals, over slides that get one going—make all the difference. While some may play fearless leaders here and there, it is nevertheless a collective effort amongst people of very-high good will, mutual respect, and fortitude to press forward. John [Legget], Janet [Walker], Rich [Furuta], as well as Ed Fox, and many others deserve great credit."

As the SIG grew, it could no longer maintain such close ties. It began to serve a larger group in a more formal way, and it had to comply with the requirements of its parent organization ACM and make choices that would affect conferences, membership, and perceptions of what SIGLINK/SIGWEB was about. Budgets, logistics, and processes of conference submissions and selection influenced SIGWEB interactions. Some hypertext-related research has not been included or disseminated within SIGWEB since the conferences can be highly selective.[64] Web-related research became a diverse field that spanned many professional groups and venues.

The history of SIGWEB and its conceptual linkages hopefully adds to the overall history of ACM. It offers a possible model for a SIG history that outlines how organizations expand and redefine their scope and how they adjust as research fields and vocabularies change. Such a model that combines both organizational and discursive dimensions can also be applicable beyond the history of a particular organization. Coupled with theories of professionalism, institutional development, and conceptual change, it provides a framework of co-development of organizations and research fields as an evolution of epistemic communities and their communications. Such a framework can serve a dual purpose of guiding studies of professional organizations and providing practical help in the analysis of gaps and pathways to success in organizations.

64. Bernstein, Everything is intertwingled: Asking again.

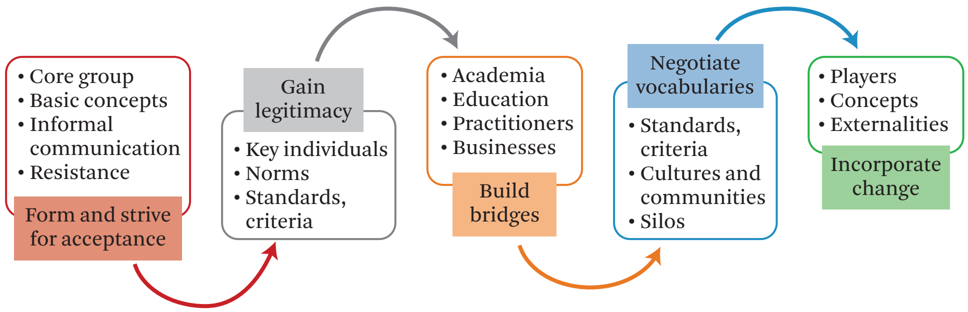

Figure 13.2 **A framework of epistemic work in a professional association.**

A professional organization exists in the nexus of multiple communities and the logics of science, market, and bureaucracy. The success of an organization depends on its abilities to negotiate common interests, align discourses and vocabularies, and establish forms of authority and expertise. In other words, each organization depends on epistemic work of its members, or the work of creating, maintaining and distributing the collective base of knowledge and practices.[65] The steps needed in such epistemic work throughout the professional organization's lifecycle can be presented as follows (Figure 13.2).

According to this framework, the epistemic work begins when a group forms around a set of concepts or common interests. The group strives for acceptance in the larger community by promoting its ideas, arguing for new possibilities via informal communications, and overcoming resistance. The next step is to gain legitimacy, which can be done by appealing to established researchers in the field, differentiating the group from other groups, and developing norms and standards that will help to define in-group/out-group boundaries. Once the group has formed and established its interests, alliances, and conceptual boundaries, it needs to build bridges, i.e., to create connections to the communities that in the future can supply human, financial, and other resources. Connections to academic and educational institutions as well as to businesses and communities of practitioners are important.

65. S. D. N. Cook and J. S. Brown. 1999. Bridging epistemologies: The generative dance between organizational knowledge and organizational knowing. *Org. Sci.*, 10(4): 381–400; M. Nerland and B. Karseth. 2015. The knowledge work of professional associations: Approaches to standardisation and forms of legitimisation. *J. Educ. Work*, 28(1): 1–23.

The next two steps include vocabulary negotiation and incorporation of change. These steps are particularly important when the group becomes established and caters to multiple research communities. Diversity of epistemic practices and forms of validity and expertise can be a source of innovation—or contention. As the group strives for cohesiveness, it may need to re-consider its standards and criteria and decide whether varying research cultures can successfully co-exist or become siloed and excluded. Studies of interdisciplinarity demonstrate the complexity of such considerations and the need to reflect on the issues of knowledge ownership, modes of production, and epistemological and technological uncertainty.[66]

The need to incorporate change is also crucial in contemporary contexts, as modern societies undergo rapid changes. These may concern changes in research concepts and directions, changing interests of new generations of researchers and practitioners, or changes in external factors, such as funding or publication models. To be incorporated, change needs to be “theorized”, i.e., specified and elaborated within the chains of cause and effect that explain new practices and their outcomes.[67]

In the case of SIGWEB, the framework suggests the following. The group was successful in forming and gaining legitimacy. It built many bridges across academia, but it somewhat neglected connections to practitioners when it decided to delay its engagement with the Web research community. It also did not succeed in its attempts to consolidate educational initiatives and curriculum-building activities. One crucial gap in the group’s evolution is a lack of visibility of standardization and certification work. For example, E. James Whitehead and others have been involved in the development of WebDAV, an extension to the HTTP protocol.[68] While this work was presented at HT conferences, it did not receive prominence as part of SIGWEB activities. Later on vocabulary re-negotiation helped the group to become more inclusive of new research areas and to stabilize and even gain membership. SIGWEB also succeeded in theorizing the change of its name. In the future,

66. M. Strathern. 2004. *Commons and Borderlands: Working Papers on Interdisciplinarity, Accountability and the Flow of Knowledge*. Sean Kingston Publishing, Oxford, UK.

67. R. Greenwood, C. R. Hinings, and R. Suddaby. 2002. Theorizing change: The role of professional associations in the transformation of institutionalized fields. *Acad. Manage. J.*, 45(1): 58–80.

68. S. Kim, M. Slater, and E. J. Whitehead, Jr. 2004. WebDAV-based hypertext annotation and trail system. In *Proceedings of the Fifteenth ACM Conference on Hypertext and Hypermedia–Hypertext’04*, pp. 87–88. ACM Press, New York.

the group may need stronger reflections on how to incorporate other changes and remain relevant to its audiences.

Understanding how organizations and research domains co-evolve is a rich and challenging area of research. This chapter takes a first step in addressing this challenge by providing a detailed account of one organization and highlighting some of its epistemic work. It also proposes a framework of epistemic work in professional organizations that can be used to transcend the specifics of one organization and to explore organizational and conceptual connections in research domains. The framework can be elaborated to further specify how an organization negotiates relationships between its own units, e.g., between the ACM and its SIGs.

Future work in this area will need to rely on a combination of methodological approaches, including comparative and ethnographic approaches and data mining and network analysis. Both the history of larger organizations, such as ACM, and the framework will benefit from comparative studies to investigate how different SIGs engage in epistemic work, negotiate their position within ACM, and engage in legitimization of expertise and standardization and certification efforts (see Chapters 2 and 5 in this volume). Ethnographic approaches will help to better capture informal communications, particularly during specialized events such as conferences. Quantitative studies of research networks and terms can help to trace changing connections between individuals and semantics within the research fields. Findings and theorizations of this chapter build the groundwork for further explorations in this area.

Acknowledgments

This research was funded in part by a 2011 ACM History Committee fellowship. The author is deeply grateful to Robert Akscyn and Mark Bernstein for their encouragement, support, and feedback on the chapter.

References

E. J. Aarseth. 1997. *Cybertext: Perspectives on Ergodic Literature*. Johns Hopkins University Press, Baltimore.

J. Abbate. 1999. *Inventing the Internet*. MIT Press, Cambridge, MA.

J. Abbate. 2012. *Recoding Gender: Women's Changing Participation in Computing*. MIT Press, Cambridge, MA.

D. H. Abernathy, J. S. Mancino, C. R. Pearson, and D. C. Swiger. 1973. Survey of design goals for operating systems. *Operat. Syst. Rev.*, 7(2): 19–34. DOI: 10.1145/775260.775263.

A. Akera. 2007. Edmund Berkeley and the origins of ACM. *Commun. ACM*, 50(5): 30–35. DOI: 10.1145/1230819.1230835.

A. Akera. 2008. *Calculating a Natural World: Scientists, Engineers, and Computers during the Rise of US Cold War Research*. MIT Press, Cambridge, MA.

G. Alberts and E. G. Daylight. 2014. Universality versus locality: The Amsterdam style of Algol implementation. *IEEE Ann. Hist. Comput.*, 36(4): 52–63. DOI: 10.1109/MAHC.2014.61.

F. L. Alt. 1962. Fifteen years of ACM. *Commun. ACM*, 5(6): 300–307. DOI: 10.1145/367766.367777.

The American Association of University Women. 1992. *How Schools Shortchange Girls: A Study of Major Findings on Girls and Education.* AAUW Educational Foundation, Washington, DC.

R. E. Anderson. 1981. EDP auditing in the 1980s: Or the vanishing paper trail. *SIGSAC Rev.*, 1(1): 6–15. DOI: 10.1145/1317419.1317420.

R. E. Anderson. 1999. SIGCAS history: The early years. *SIGCAS Comput. Soc.*, 29(4): 4–5. DOI: 10.1145/572199.572200.

H. Arendt. 1966. On the human condition. In A. M. Hilton and Institute for Cybercultural Research, editors. *The Evolving Society: The Proceedings of the First Annual Conference* *on the Cybercultural Revolution—Cybernetics and Automation*, pp. 213–220. Institute for Cybercultural Research, New York.

C. Arnst and A. Dooley. July 23, 1975. Majority of women DPers find no job discrimination. *Computerworld*, pp. 1–4.

W. Aspray and B. O. Williams. 1994. Arming American scientists: NSF and the provision of scientific computing facilities for universities, 1950–1973. *IEEE Ann. Hist. Comput.*, 16(4): 60–74.

W. Aspray. 1986. International diffusion of computer technology, 1945–1955. *IEEE Ann. Hist. Comput.*, 8(4): 351–360. DOI: 10.1109/MAHC.1986.10063.

W. Aspray. 2000. Was early entry a competitive advantage? US universities that entered computing in the 1940s. *IEEE Ann. Hist. Comput.*, 22(3): 42–87. DOI: 10.1109/85.859525.

W. Aspray. 2014. Michael Sean Mahoney (1939–2008). *IEEE Ann. Hist. Comput.*, 36(3): 70–79.

W. Aspray. 2016. *Participation in Computing: The National Science Foundation's Expansionary Programs*. Springer International, Basel.

W. Asprey and W. Laffan. 1975. Women speak out on DP careers. *Datamation*, 21(8): 41–44.

W. F. Atchison and J. W. Hamblen. 1964. Status of computer sciences curricula in colleges and universities. *Commun. ACM*, 7(4): 225–227.

W. F. Atchison. 1981. Computer education, past, present, and future. *SIGCSE Bull.*, 13(4): 2–6. DOI: 10.1145/989306.989307.

W. F. Atchison, S. D. Conte, J. W. Hamblen, T. E. Hull, T. A. Keenan, W. B. Kehl, E. J. McCluskey, S. O. Navarro, W. C. Rheinboldt, E. J. Schweppe, W. Viavant, and D. M. Young, Jr. 1968. Curriculum 68: Recommendations for academic programs in computer science: A report of the ACM Curriculum Committee on Computer Science. *Commun. ACM*, 11(3): 151–197. DOI: 10.1145/362929.362976.

L. Bailyn. 2003. Academic careers and gender equity: Lessons learned from MIT. *Gender, Work Org.*, 10(2): 137–153. DOI: 10.1111/1468–0432.00008.

V. Balasubramanian, M. Bieber, and T. Isakowitz. 1996. *Systematic Hypermedia Design*. Retrieved from http://archive.nyu.edu/bitstream/2451/14198/1/IS-96–09.pdf

D. Banta. 2013. Inspiration lost and found. *ACM-W CIS Newslett.*, 4(2): 22–24.

J. T. Barber. 2006. *The Black Digital Elite*. Praeger, Westport, CT.

T. Bardini. 2000. *Bootstrapping: Douglas Engelbart, Coevolution, and the Origins of Personal Computing.* Stanford University Press, Stanford.

T. Bardini. 2006. Bridging the gulfs: From hypertext to cyberspace. *J. Comput.-Mediat. Commun.* 3(2). DOI: 10.1111/j.1083–6101.1997.tb00069.

M. Bariff. 1981. DP control and standards activities. *SIGSAC Rev.*, 1(1): 20–25.

L. J. Barker and W. Aspray. 2006. The state of research on girls and IT. In J. McGrath Cohoon and W. Aspray, editors. *Women and Information Technology: Research on Underrepresentation*, pp. 3–54. MIT Press, Cambridge, MA. DOI: 10.7551/mitpress/9780262033459.003.0001.

B. Barnet. 2013. *Memory Machines: The Evolution of Hypertext*. Anthem Press, London.

E. Barrett. 1991. *The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information*. MIT Press, Cambridge, MA.

M. Bauman. May 14–16, 1969. Computers and the underprivileged. *AFIPS '69 (Spring) Proceedings of the May 14–16, 1969, Spring Joint Computer Conference*, pp. 35–36. ACM, New York.

D. O. Belanger. 1998. *Enabling American Innovation: Engineering and the National Science Foundation*. Purdue University Press, West Lafayette, IN.

D. Bellin. 1985. High school hackers: Heroes or criminals? *Comput. Soc.*, (14 & 15): 16–17. DOI: 10.1145/379486.379507.

S. Berenson, et al. 2004. Voices of women in a software engineering course: Reflections on collaboration. *ACM J. Educ. Resourc. Comput.*, 4(1): 1–18.

T. J. Bergin. 2007. A history of the history of programming languages. *Commun. ACM*, 50(5): 69–74. DOI: 10.1145/1230819.1230841.

T. Berners-Lee and M. Fischetti. 1999. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. Harper, San Francisco.

A. J. Bernstein. 1971. A programmer training project. *Commun. ACM*, 14(7): 449–452. DOI: 10.1145/362619.362624.

M. Bernstein. 1999. Everything is intertwingled: Asking again. In *Proceedings of the ACM Conference on Hypertext - Hypertext'99*, ACM Press. Retrieved from www.eastgate.com/ht99/slides/Welcome.htm

K. W. Beyer. 2009. *Grace Hopper and the Invention of the Information Age*. MIT Press, Cambridge, MA.

A. S. Bix. 2013. *Girls Coming to Tech! A History of American Engineering Education for Women*. MIT Press, Cambridge, MA.

A. Blackwell and K. Rodden. 2003/1963. Preface to this electronic edition. In *Sketchpad: A Man-machine Graphical Communication System*. Cambridge University Online.

H. Blinn. 1998. Ten more unsolved problems in computer graphics. *IEEE Comput. Graph. Applic.*, 18(5): 86–89. DOI: 10.1109/38.708564.

B. Bloombecker. 1987. Computer ethics: An antidote to despair. *Comput. Soc.*, 16(4): 3–11. DOI: 10.1145/25601.25602.

B. (J.) Bloombecker. 1983. Conscience in computing: A Law Day perspective on computer crime. *Comput. Soc.*, 13(3): 9–13. DOI: 10.1145/958600.958603.

B. (J.) Bloombecker. 1990. Needed: Binary Bar Mitzvah's and computer confirmations. *Comput. Soc.*, 20(4): 20–21. DOI: 10.1145/122403.122408.

A. Bochannek. 2011. What did the Americans know? A review of select periodicals. In A. N. Tomilin, editor. 2011. *SoRuCom–2011 Proceeding, Second International Conference on the History of Computers and Informatics in the Soviet Union and Russian Federation*, Veliky Novgorod, pp. 3–7.

J. D. Bolter. 2001. *Writing Space: Computers, Hypertext, and the Remediation of Print*. Lawrence Erlbaum Associates, Mahwah, NJ.

J. D. Bolter and M. Joyce. 1987. Hypertext and creative writing. In *Proceedings of the ACM Conference on Hypertext–HYPERTEXT '87*, pp. 41–50. ACM Press, New York. DOI: 10.1145/317426.317431.

G. Bowker. 1993. How to be universal: Some cybernetic strategies, 1943–70. *Soc. Stud. Sci.*, 23(1): 107–127.

C. P. Brown. 1972. An investigation into the recruitment and training of the disadvantaged for staffing the MIS function. *Comput. Personn. Quart. Public. Spec. Inter. Group Comput. Personn. Res.*, 3(2): 3–8. DOI: 10.1145/1036396.1036397.

J. J. Brown. 1967. *Ideas in Exile: A History of Canadian Invention.* McClelland and Stewart Ltd., Toronto.

J. Brown and S. Cunningham. 2007. A history of ACM SIGGRAPH. *Commun. ACM*, 50(5): 54–61. DOI: 10.1145/1230819.1230839.

P. Brown, C. Gere, N. Lambert, and C. Mason. 2009. *White Heat Cold Logic: Early British Computer Art 1960–1980*. MIT Press, Cambridge, MA.

K. B. Bruce, R. D. Cupper, and R. L. S. Drysdale. 2010. A history of the liberal arts computer science consortium and its model curricula. *ACM Trans. Comput. Educ.*, 10(1): 3:1–3:12. DOI: 10.1145/1731041.1731044.

C. Burck, editor. July 1968. Employing the unemployable. *Fortune*, 78(1): 29–30, 34.

J. H. Burrows, Chairman. May 14–16, 1969. A panel session: Computers and the underprivileged. *AFIPS '69 (Spring) Proceedings of the May 14–16, 1969, Spring Joint Computer Conference*, pp. 35–40. ACM, New York.

V. Bush. July 1945. As we may think. *The Atlantic*, pp. 101–108. Available at www.theatlantic.com/magazine/archive/1945/07/as-we-may-think/303881/

T. Camp. 1997. The incredible shrinking pipeline. *Commun. ACM*, 40(10): 103–110.

T. Camp. Response. 1998. *Commun. ACM*, 41(3): 27–28.

S. M. Campbell. 2009. Backwater calculations for the St. Lawrence Seaway and the first computer in Canada. *Canadian Journal of Civil Engineering*, 36(7): 1164–1169.

S. M. Campbell. 2003. Beatrice Helen Worsley: Canada's female computer pioneer. *IEEE Ann. Hist. Comput.*, 25(3): 51–62.

S. M. Campbell. 2006. The premise of computer science: Establishing modern computing at the University of Toronto (1945–1964). Ph.D. thesis, Institute for the History and Philosophy of Science and Technology, University of Toronto.

M. Campbell-Kelly and W. Aspray. 1996. *Computer: A History of the Information Machine*. Basic Books, New York.

M. Campbell-Kelly. 1985. Christopher Strachey, 1916–1975: A biographical note. *Ann. Hist. Comput.*, 7(1): 19–42. DOI: 10.1109/MAHC.1985.10001.

R. P. Canavan. 1986. ACM Forum: Reported hacking. *Commun. ACM*, 29(7): 590.

L. Candela, D. Castelli, and P. Pagano. 2011. History, evolution, and impact of digital libraries. In I. Iglezakis, T. Synodinou, and S. Kapidakis, editors. *E-Publishing and Digital Libraries: Legal and Organizational Issues*, pp. 1–30. Information Science Reference, Hershey, PA. DOI: 10.4018/978–1-60960–031-0.ch001.

R. J. Carnota. 2015. The beginning of computer science in Argentina and the Calculus Institute, 1957–1970. *IEEE Ann. Hist. Comput.*, 37(4): 40–52.

S. J. Ceci and W. M. Williams, editors. 2007. *Why Aren't More Women in Science?* American Psychological Association, Washington, DC.

P. E. Ceruzzi. 1998. *A History of Modern Computing*. MIT Press, Cambridge, MA.

M. Charland. 1986. Technological nationalism. *Canadian Journal of Political and Social Theory*, 10(1): 196–220.

D. Clark and D. Wilson. 1987. A comparison of commercial and military computer security policies. In *Proceedings of the 1987 IEEE Symposium on Security and Privacy*, pp. 184–194. Oakland, CA. DOI: 10.1109/SP.1987.10001.

A. Cochran. 1987. ACM: The past 15 years, 1972–1987. *Commun. ACM*, 30(10): 866–872. DOI: 10.1145/30408.30419.

J. M. Cohoon and W. Aspray, editors. 2006. *Women and Information Technology: Research on Underrepresentation*. MIT Press, Cambridge, MA.

J. M. Cohoon and W. Aspray. 2006. A critical review of the research on women's participation in postsecondary computing education. In J. McGrath Cohoon and William Aspray, editors. 2006. *Women and Information Technology: Research on Underrepresentation*, pp. 137–180. MIT Press, Cambridge, MA. DOI: 10.7551/mitpress/9780262033459.003.0005.

J. M. Cohoon. 1999. Departmental differences can point the way to improving female retention in computer science. In *SIGCSE '99 The Proceedings of the Thirtieth SIGCSE Technical Symposium on Computer Science Education*, pp. 198–202. ACM, New York. DOI: 10.1145/299649.299753.

D. E. Comer, D. Gries, M. C. Mulder, A. Tucker, A. J. Turner, and P. R. Young. 1989. Computing as a discipline. *Commun. ACM*, 32(1): 9–23. DOI: 10.1145/63238.63239.

J. Conklin. 1987. Hypertext: An introduction and survey. *IEEE Comput.*, 20(9): 17–41. DOI: 10.1109/MC.1987.1663693.

S. D. N. Cook and J. S. Brown. 1999. Bridging epistemologies: The generative dance between organizational knowledge and organizational knowing. *Org. Sci.*, 10(4): 381–400. DOI: 10.1287/orsc.10.4.381.

C. Corbett and C. Hill. 2015. *Solving the Equation: The Variables for Women's Success in Engineering and Computing.* AAUW, Washington DC.

J. W. Cortada. 2012. *The Digital Flood: The Diffusion of Information Technology across the U.S., Europe, and Asia.* Oxford University Press, New York.

D. Crawford. 1989. From Washington: Two bills equal forewarning. *Commun. ACM*, 32(7): 780–782.

H. Crowther-Heyck. 2005. *Herbert A. Simon: The Bounds of Reason in Modern America*. Johns Hopkins University Press, Baltimore.

W. E. Cundiff and M. Reid, editors. 1979. *Issues in Canadian/US Transborder Computer Data Flows.* Toronto: Institute for Research on Public Policy. DOI: 10.1145/959310.959312.

G. Davida, D. L. Wells, and J. B. Kam. 1981. A database encryption system with subkeys. *Trans. Database Syst.*, 6(2): 312–328. DOI: 10.1145/319566.319580.

C.-S. Davis, A. B. Ginorio, C. S. Hollenshead, B. B. Lazarus, and P. M. Rayman. 1996. *The Equity Equation: Fostering the Advancement of Women in the Sciences, Mathematics, and Engineering*. Jossey-Bass Publishers, San Francisco.

P. Davis. January 8, 1998. Remembering George Forsythe. *SIAM News*. Available at http://www.siam.org/news/news.php?id=805.

W. A. DeLegall. 1969. Teaching techniques and quality education/training for the 'Disadvantaged'. *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*, pp. 91–94. ACM, New York. DOI: 10.1145/800163.805184.

L. DeNardis. 2015. The Internet design tension between surveillance and security. *IEEE Ann. Hist. Comput.*, 37(2): 72–83. DOI:10.1109/MAHC.2015.29.

D. E. Denning. 1976. A lattice model of secure information flow. *Commun. ACM*, 19(5): 236–243.

D. E. Denning. 1980. Secure statistical databases with random sample queries. *Trans. Database Syst.*, 5(3): 291–315. DOI: 10.1145/320613.320616.

D. E. Denning. 1991. The United States vs. Craig Neidorf: A debate on electronic publishing, Constitutional rights, and hacking. *Commun. ACM*, 34(3): 22–43. DOI: 10.1145/102868.102869.

D. E. Denning and P. J. Denning. 1979. The tracker: A threat to statistical database security. *Trans. Database Syst.*, 4(1): 76–96. DOI: 10.1145/320064.320069.

P. J. Denning. 1977. Discrimination and affirmative action. *Commun. ACM*, 20(3): 198.

P. J. Denning. 1988. Edmund C. Berkeley—ACM founder. *Commun. ACM*, 31(6): 781–782. DOI: 10.1145/62959.214901.

P. J. Denning. 1990. Editorial: Sending a signal. *Commun. ACM*, 33(8): 11–13.

P. J. Denning and C. H. Martell. 2015. *Great Principles of Computing.* MIT Press, Cambridge, MA.

P. J. Denning and H. S. Stone. 1980. An exchange of views on operating systems courses. *Operat. Syst. Rev.*, 14(4): 71–82. DOI: 10.1145/850708.850714.

P. J. Denning, D. H. Brandin, D. C. Schwartz, and G. I. Davida. 1981. Report of the public cryptography study group. *Commun. ACM*, 24(7): 434–450. DOI: 10.1145/358699.358707.

J. B. Dennis. 1967. A position paper on computing and communications. In *SOSP '67 Proceedings of the First ACM Symposium on Operating System Principles*, 6.1–6.10. DOI: 10.1145/363095.363147.

L.-B. Desroches, et al. December 1, 2014. Computer usage and national energy consumption: Results from a field-metering study, Lawrence Berkeley National Laboratory, December 1. Available at http://eetd.lbl.gov/sites/all/files/computers_lbnl_report_v4.pdf.

W. Diffie and M. E. Hellman. 1981. On the security of multiple encryption. *Commun. ACM*, 24(7): 465–467.

W. Diffie and M. Hellman. June 7–10, 1976. Multiuser cryptographic techniques. In *AFIPS '76 Proceedings of the June 7–10, 1976, National Computer Conference*, pp. 109–112. DOI: 10.1145/1499799.1499815.

T. A. DiPrete and C. Buchmann. 2013. *The Rise of Women: The Growing Gender Gap in Education and What it Means for American Schools*. Russell Sage Foundation, New York.

S. Donig. 2010. Appropriating American technology in the 1960s: Cold War politics and the GDR computer industry. *IEEE Ann. Hist. Comput.*, 32(2): 32–45.

H. Durnova. 2014. Embracing the Algol effort in Czechoslovakia. *IEEE Ann. Hist. Comput.*, 36(4): 26–37. DOI: 10.1109/MAHC.2014.51.

S. Dziallas and S. Fincher. 2015. ACM Curriculum Reports: A pedagogic perspective. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, pp. 81–89. ACM, New York. DOI: 10.1145/2787622.2787714.

P. N. Edwards. 1996. *The Closed World: Computers and the Politics of Discourse in Cold War America*. MIT Press, Cambridge, MA.

G. Engel, J. Impagliazzo, and P. LaMalva. 2009. A brief history of the Computing Sciences Accreditation Board (CSAB): Promoting quality education in the computing fields. In *2009 IEEE Conference on the History of Technical Societies*, pp. 1–8. DOI: 10.1109/HTS.2009.5337839.

D. C. Engelbart. 1986. The augmented knowledge workshop. In *Proceedings of the ACM Conference on the History of Personal Workstations*, pp. 73–83. ACM Press, New York. DOI: 10.1145/12178.12184.

D. C. Engelbart. 1995. Toward augmenting the human intellect and boosting our collective IQ. *Commun. ACM*, 38(8): 30–32. DOI: 10.1145/208344.208352.

N. L. Ensmenger. 2001. The 'Question of Professionalism' in the computer fields. *IEEE Ann. Hist. Comput.*, 23(4): 1–19. DOI: 10.1109/85.969964.

N. L. Ensmenger. 2012. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise.* MIT Press, Cambridge, MA.

N. L. Ensmenger. 2015. 'Beards, sandals, and other signs of rugged individualism': Masculine culture within the computing professions. *Osiris*, 30(1): 38–65. DOI: 10.1086/682955.

A. P. Ershov. 1963. *Input Language for Automatic Programming Systems*. Academic Press, New York.

A. P. Ershov. 1960. Predislovie redaktora perevoda. In P. Naur, editor. *Soobshchenie ob algoritmicheskom iazyke ALGOL 60*, Moscow/Novosibirsk: Vychislitel'nyi Tsentr AN SSSR, pp. 3–9.

A. P. Ershov. 1965. The Alpha—An automatic programming system of high efficiency. In *Proceedings of IFIP Congress 65*, volume 2, pp. 622–623. Macmillan and Co., London. DOI: 10.1145/321312.321314.

A. P. Ershov. 1959. *Programming Programme for the BESM Computer*. Pergamon Press, London.

A. P. Ershov and A. A. Liapunov. 1967. O formalizatsii poniatiia programmy. *Kibernetika*, (5): 40–57.

A. P. Ershov, S. S. Lavrov, and M. R. Shura-Bura. 1965. In P. Naur, editor. *Algoritmicheskii iazyk ALGOL 60: Peresmotrennoe soobshchenie.* Mir, Moscow.

N. Erskine. December 1970. A submission by the Canadian Information Processing Society related to the subject of participation by telecommunication carriers in public data processing. *CIPS Mag.*, 1(11): 10–13.

U. Feest and F. Steinle, editors. 2012. *Scientific Concepts and Investigative Practice*. Walter de Gruyter, Berlin.

E. A. Feigenbaum. May 1965. From Minsk to Pinsk without Intourist. *Datamation*, 11(5): 34–36.

E. A. Feigenbaum. 1961. Soviet cybernetics and computer sciences, 1960. *Commun. ACM*, 4(12): 566–579. DOI: 10.1145/366853.366891.

L. Fein. 1963. Computer-oriented peace research. *AFIPS '63 (Fall) Proceedings of the November 12–14, 1963*, 631–639. DOI: 10.1145/1463822.1463890.

L. Fein. June 1961. The computer-related sciences (Synnoetics) at a university in the year 1975, *Am. Sci.*, 49(2): 149–168. DOI: 10.1145/364005.364020.

L. Fein. 1959. The role of the university in computers, data processing, and related fields. *Commun. ACM*, 2(9): 7–14. DOI: 10.1145/1457838.1457859.

W. Fetter. 1982. A progression of human figures simulated by computer graphics. *IEEE Comput. Graph. Applic.*, 2(9): 9–13. DOI: 10.1109/MCG.1982.1674468.

J. Foley. 2000. Getting there: The ten top problems left. *IEEE Comput. Graph. Applic.*, 20(1): 66–68. DOI: 10.1109/38.814569.

K. A. Forcht. 1986. The need for including data security topics in the college business curriculum. *SIGSAC Rev.*, 4(3): 9–11. DOI: 10.1145/1058414.1058416.

D. E. Forsythe. 2001. George and Sandra's daughter, Robert's girl: Doing ethnographic research in my parents' field. In D. J. Hess, editor. *Studying Those Who Study Us: An Anthropologist in the World of Artificial Intelligence*, pp. 183–196. Stanford University Press, Stanford.

D. E. Forsythe. 2001. Disappearing women in the social world of computing. In D. J. Hess, editor. *Studying Those Who Study Us: An Anthropologist in the World of Artificial Intelligence*, pp. 163–182. Stanford University Press, Stanford.

G. E. Forsythe. June 25, 1965. Stanford University's program in computer science. Technical Report CS26. Computer Science Department, Stanford University. Available at http://i.stanford.edu/pub/cstr/reports/cs/tr/65/26/CS-TR-65-26.pdf

G. E. Forsythe. 1967. A university's educational program in computer science. *Commun. ACM*, 10(1): 3–11. DOI: 10.1145/363018.363038.

G. E. Forsythe. 1965. Official ACM. *Commun. ACM*, 8(7): 422–426.

G. E. Forsythe. 1968. What to do till the computer scientist comes? *Am. Math. Month.*, 75(5): 454–462. DOI: 10.2307/2314698.

G. Forsythe and W. Wasow. 1960. *Finite-Difference Methods for Partial Differential Equations.* Wiley, New York.

G. Forsythe. 1965. President's letter to the ACM membership: Why ACM? *Commun. ACM*, 8(3): 143–144. DOI: 10.1145/364995.364997.

E. A. Fox, R. M. Akscyn, R. K. Furuta, and J.J. Leggett. 1995. Digital libraries. *Commun. ACM*, 38(4): 22–28. DOI: 10.1145/205323.205325.

R. D. Francis. 2010. *The Technological Imperative in Canada.* UBC Press, Vancouver.

K. Frenkel. 1990. Women and computing. *Commun. ACM*, 33(11): 34–46.

J. Gaboury. 2015. Hidden surface problems: On the digital image as material object. *J. Vis. Cult.*, 14(1): 40–60. DOI: 10.1177/1470412914562270.

R. S. Gaines. 1975. Session 3: Protection. *Operating Systems Review*, 9(3): 58. DOI: 10.1145/800272.810900.

R. S. Gaines and D. K. Hsiao. 1975. Preface. In *Proceedings of the 1st International Conference on Very Large Data Bases*.

C. Gallagher. 1970. The computer as Rosemary's baby. *Comput. Soc.*, 1(2): 4.

C. Galvan. 2012. What can we learn from computer engineer Barbie? *ACM-W CIS Newsletter*, 4(1): 21–32.

K. Garvin-Doxas and L. J. Barker. 2004. Communication in computer science classrooms: Understanding defensive climates as a means of creating supportive behaviors. *ACM J. Educ. Resour. Comput.*, 4(1): 1–18. DOI: 10.1145/1060071.1060073.

S. Gerovitch. 2001. *From Newspeak to Cyberspeak: A History of Soviet Cybernetics*. MIT Press, Cambridge, MA.

E. M. Gerson. 1977. Directory of researchers in computers and society. *Comput. Soc.*, 8(4): 5–9. DOI: 10.1145/958901.958903.

N. E. Gibbs and A. B. Tucker. 1986. A model curriculum for a liberal arts degree in computer science. *Commun. ACM*, 29(3): 202–210. DOI: 10.1145/5666.5667.

T. F. Gieryn. 1999. *Cultural Boundaries of Science: Credibility on the Line*. University of Chicago Press, Chicago.

J. P. Gilbert and D. B. Mayer. 1969. Experiences in self-selection of disadvantaged people into a computer operator training program. *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*, pp. 79–90. ACM, New York. DOI: 10.1145/800163.805183.

C. S. Gillmor. 2004. *Fred Terman at Stanford: Building a Discipline, a University, and Silicon Valley.* Stanford University Press, Stanford.

C. Ginzburg. 1980. *The Cheese and the Worms: The Cosmos of a Sixteenth-Century Miller*. Johns Hopkins University Press, Baltimore.

E. F. Glynn. 1998. More studies needed. *Commun. ACM*, 41(3): 27.

A. Goldberg. 1985. Reliability of computer systems and risks to the public. *Commun. ACM*, 28(2): 131–133.

M. L. Goldberger, B. A. Maher, and P. E. Flattau, editors. 1995. *Research Doctorate Programs in the United States: Continuity and Change*. National Academy Press, Washington, DC.

G. A. Good. 2000. The assembly of geophysics: Scientific disciplines as frameworks of consensus. *Stud. Hist. Philoso. Sci. (Part B)*, 31(3): 259–292. DOI: 10.1016/S1355-2198(00)00018-6.

S. Gorn. 1963. The computer and information sciences: A new basic discipline. *SIAM Rev.*, 5(2): 150–155. DOI: 10.1137/1005036.

C. M. Gorriz and C. Medina. 2000. Engaging girls with computers through software games. *Commun. ACM*, 43(1): 42–49. DOI: 10.1145/323830.323843.

R. Graham. 1967. Protection in an information utility. In *SOSP '67: Proceedings of the First ACM Symposium on Operating System Principles*. DOI: 10.1145/800001.811674.

V. Green. 2001. *Race on the Line: Gender, Labor, and Technology in the Bell System, 1880–1980*. Duke University Press, Durham, NC.

R. Greenwood, C. R. Hinings, and R. Suddaby. 2002. Theorizing change: The role of professional associations in the transformation of institutionalized fields. *Acad. Manage. J.*, 45(1): 58–80. DOI: 10.2307/3069285.

D. A. Grier. 2005. *When Computers were Human*. Princeton University Press, Princeton.

H. Griffin and P. Gravelle. 1969. Report on the philosophy and mechanics of the urban education committee of Philadelphia. *SIGCPR '69 Proceedings of the Seventh Annual Conference on SIGCPR*, pp. 95–98. ACM, New York. DOI: 10.1145/800163.805185.

K. Grønbæk and R. H. Trigg. 1999. *From Web to Workplace: Designing Open Hypermedia Systems*. MIT Press, Cambridge, MA.

G. K. Gupta. 2007. Computer science curriculum developments in the 1960s. *IEEE Ann. Hist. Comput.*, 29(2): 40–54.

I. Hacking. 1999. *The Social Construction of What?* Harvard University Press, Cambridge, MA.

T. Haigh. 2001. The chromium-plated tabulator: Institutionalizing an electronic revolution, 1954–1958. *IEEE Ann. Hist. Comput.*, 23(4): 75–104. DOI: 10.1109/85.969965.

T. Haigh. 2002. Technology, information and power: Managerial technicians in corporate America, 1917–2000. Ph.D. Dissertation, University of Pennsylvania.

T. Haigh. 2008. Cleve Moler: Mathematical software pioneer and creator of Matlab. *IEEE Ann. Hist. Comput.*, 30(1): 87–91. DOI: 10.1109/MAHC.2008.2.

T. Haigh. 2010. Masculinity and the masculine man. In T. J. Misa, editor. *Gender Codes: Why Women Are Leaving Computing*, pp. 51–71. Wiley IEEE Computer Society, Hoboken, NJ.

T. Haigh, E. Kaplan, and C. Seib. 2007. Sources for ACM history. *Commun. ACM*, 50(5): 36–41. DOI: 10.1145/1230819.1230836.

T. Haigh, M. Priestley, and C. Rope. 2016. *ENIAC in Action: Making and Remaking the Modern Computer*. MIT Press, Cambridge, MA.

F. G. Halasz. 1987. Reflections on NoteCards: Seven issues for the next generation of hypermedia systems. In *Proceeding of the ACM Conference on Hypertext–HYPERTEXT '87*, pp. 345–365. ACM Press, New York. DOI: 10.1145/317426.317451.

F. G. Halasz and M. Schwartz. 1994. The Dexter hypertext reference model. *Commun. ACM*, 37(2): 30–39. DOI: 10.1145/175235.175237

M. Hall. 2012. Understanding ACM's past. *Commun. ACM*, 55(12): 5. DOI: 10.1145/2380656.2380657.

W. Hall. 2007. Back to the future with Hypertext: A tale of two or three conferences. In *Proceedings of the Eighteenth Conference on Hypertext and Hypermedia*, (HT '07), pp. 179–180. ACM, New York. DOI: 10.1145/1286240.1286293.

D. Haltrect. 1970. Computer show/competition. *CIPS Mag.*, 1(7): 10.

R. Hammerman and A. L. Russell, editors. 2015. *Ada's Legacy: Cultures of Computing from the Victorian to the Digital Age*. Association for Computing Machinery and Morgan & Claypool, New York. DOI: 10.1145/2809523.

V. Hanson, R. Ayfer, and B. Bachmayer. 2014. European women in computing. *Commun. ACM*, 57(7): 5.

T. R. Harbron. 1984. More on accreditation. *Commun. ACM*, 27(10): 988. DOI: 10.1145/358274.358276.

J. Hartmanis. 1992. Computing the future. *Commun. ACM*, 35(11): 31–40.

J. Hartmanis. 1994. On computational complexity and the nature of computer science. *Commun. ACM*, 37(10): 37–43. DOI: 10.1145/214037.214040.

J. Hartmanis and H. Lin, editors. 1992. *Computing the Future: A Broader Agenda for Computer Science and Engineering*. National Academy Press, Washington, DC.

C. C. Hayes. 2010. Computer science: The incredible shrinking woman. In T. J. Misa, editor. *Gender Codes: Why Women are Leaving Computing*, pp. 25–49. Wiley IEEE Computer Society, New York. DOI: 10.1002/9780470619926.ch2.

R. V. Head. 2004. *Datamation*'s glory days. *IEEE Ann. Hist. Comput.*, 26(2): 16–21.

P. Heckbert. April 1987. Ten unsolved problems in rendering. In *Workshop on Rendering Algorithms and Systems, Graphics Interface 87*, Toronto.

E. B. Heinlein. 1989. ACM Forum: Beyond worms. *Commun. ACM*, 32(6): 674.

V. Henriques. 1970. Computing and the disadvantaged, a panel session. *ACM '70: Proceedings of the 1970 25th Annual Conference on Computers and Crisis: How Computers Are Shaping Our Future*, pp. 8–15. ACM, New York. DOI: 10.1145/1147347.1147286.

M. J. Heron, P. Belford, and A. Goker. 2014. Sexism in the circuitry: Female participation in male-dominated popular computer culture. *ACM SIGCAS Comput. Soc.*, 44(4): 18–29. DOI: 10.1145/2695577.2695582.

R. Hersch. 1992. In memoriam: Nicholas D. Kazarinoff. *SIAM Rev.*, 34(1): 127. DOI: 10.1137/1034017.

M. Hicks. 2010. Only the clothes changed: Women operators in British computing and advertising, 1950–1970. *IEEE Ann. Hist. Comput.*, 32(2): 2–14. DOI: 10.1109/MAHC.2010.55.

H. Higgins and D. Kahn, editors. 2012. *Mainframe Experimentalism: Early Computing and the Foundations of the Digital Arts*. University of California Press, Berkeley.

C. Hill, C. Corbett, and A. St. Rose. 2010. *Why So Few?: Women in Science, Technology, Engineering, and Mathematics.* AAUW, Washington DC.

M. A. Hiltzik. 1999. *Dealers of Lightning: Xerox PARC and the Dawn of the Computer Age*. Harper Business.

L. H. Hin, A. L. Wilkinson, D. H. Anderson, D. P. Chang, W. Wright, A. J. Mayo, I. T. Viney, and R. Williams. 1981. A penetration analysis of a Burroughs large system. *Operat. Syst. Rev.*, 15(1): 14–25. DOI: 10.1145/1041454.1041455.

D. C. Hoefler. January 11, 1971. Silicon Valley in the USA. *Electronic News*.

C. Hollenshead, S. Wenzel, B. Lazarus, and I. Nair. 1996. The graduate experiences in the sciences and engineering: Rethinking a gendered institution. In C.-S. Davis, et al., editors. *The Equity Equation: Fostering the Advancement of Women in the Sciences, Mathematics, and Engineering*, pp. 122–163. Jossey-Bass Publishers, San Francisco.

D. K. Hsiao. 1980. TODS: The first three years (1976–1978). *Trans. Database Syst.*, 5(4): 385–403. DOI: 10.1145/320610.320611.

J. Hurst, M. S. Mahoney, N. H. Taylor, D. T. Ross, and R. M. Fano. July–August, 1989. Retrospectives I: The early years in computer graphics at MIT, Lincoln Lab and Harvard, and Retrospectives II: The early years in computer graphics at MIT, Lincoln Lab and Harvard. *SIGGRAPH '89 Panel Proceedings, July 31–Aug. 4, 1989*, pp. 19–73. Association for Computing Machinery, Boston/New York. DOI: 10.1145/77277.77279.

E. T. James, editor. 1971. s.v., "Nora Stanton Blatch Barney." *Notable American Women: A Biographical Dictionary*. Belknap Press of Harvard University Press, Cambridge, MA.

R. F. Johnston. 1952. The University of Toronto electronic computer. *ACM '52: Proceedings of the 1952 ACM National Meeting (Toronto)*, pp. 154–160. ACM Press, New York.

D. H. Jonassen. 1989. *Hypertext/Hypermedia*. Educational Technology Publications, Englewood Cliffs, NJ.

A. Kay. 1969. The reactive engine. University of Utah Dissertation.

E. F. Keller. 1985. *Reflections on Gender and Science.* Yale University Press, New Haven, CT.

S. Kim, M. Slater, and E. J. Whitehead, Jr. 2004. WebDAV-based hypertext annotation and Trail System. In *Proceedings of the Fifteenth ACM Confereence on Hypertext and Hypermedia—Hypertext'04*, pp. 87–88. ACM Press, New York. DOI: 10.1145/1012807.1012832.

A. Kitzmann. 2006. *Hypertext Handbook: The Straight Story*. Peter Lang, New York.

M. Klawe and N. Leveson. 1995. Women in computing: Where are we now? *Commun. ACM*, 38(1): 29–35. DOI: 10.1145/204865.204874.

R. R. Kline. 2015. *The Cybernetics Moment: Or Why We Call Our Age the Information Age*. Johns Hopkins University Press, Baltimore.

R. Kling. 1981. Computer abuse and computer crime as organizational activities. *Comput. Soc.*, 11(4): 13. DOI: 10.1145/957869.957871.

R. Kling. 1976. Social impacts of computing. *Comput. Soc.*, 7(2).

D. E. Knuth. 1972. George Forsythe and the development of computer science. *Commun. ACM*, 15(8): 721–727. DOI: 10.1145/361532.361538.

D. E. Knuth. 1974. Computer science and its relation to mathematics. *Am. Math. Month.*, 81: 323–343. DOI: 10.2307/2318994.

N. Krementsov. 2005. *International Science between the World Wars: The Case of Genetics*. Routledge, New York.

J. Krige. 2006. Atoms for Peace, scientific internationalism, and scientific intelligence. *Osiris*, 21:161–181. DOI: 10.1086/507140.

F. N. Krull. 1994. The origin of computer graphics within General Motors. *IEEE Ann. Hist. Comput.*, 16(3): 40–56. DOI: 10.1109/MAHC.1994.298419.

B. Lampson. 1973. A note on the confinement problem. *Commun. ACM*, 16(10): 613–15. DOI: 10.1145/362375.362389.

B. Lampson. 1971. Protection. *Operat. Syst. Rev.*, 8(1): 18. Reprinted from Butler Lampson, Protection. In *Proceedings of Fifth Princeton Symposium on Information Sciences and Systems*, pp. 437–443. Princeton.

B. Lampson, et al. 1977. Protection, security, reliability. *Operat. Syst. Rev.*, 11(1): 13.

C. P. Landis. 1973. Computer records and people. *Comput. Soc.*, 4(3): 15.

G. P. Landow. 2006. *Hypertext 3.0: Critical Theory and New Media in an Era of Globalization*. Johns Hopkins University Press, Baltimore.

B. F. Laposky. 1969. Oscillons: Electronic abstractions. *Leonardo*, 2(4): 345–354. DOI: 10.2307/1572117.

R. D. Launius. 1999. NASA history and the challenge of keeping the contemporary past. *Publ. Hist.*, 21(3): 63–81. DOI: 10.2307/3378961.

J. A. N. Lee, G. Segal, and R. Steier. 1986. Positive alternatives: A report on an ACM panel on hacking. *Commun. ACM*, 29(4): 297–299. DOI: 10.1145/5684.6377.

A. Leonard. June 10, 1999. Can history survive Silicon Valley? *Salon*. http://www.salon.com/1999/06/10/stanford/.

S. Levy. 2001. *Crypto: How the Code Rebels Beat the Government—Saving Privacy in the Digital Age.* Penguin Books, New York.

N. Lewis. 2016. Peering through the curtain: Soviet computing through the eyes of western experts. *IEEE Ann. Hist. Comput.*, 38(1): 34–47. DOI: 10.1109/MAHC.2015.48

A. A. Liapunov. 1958. O logicheskikh skhemakh programm. *Problemy Kibernetiki*, 1: 46–74. Fizmatgiz, Moscow.

J. C. R. Licklider. 1960. Man-computer symbiosis. *IRE Trans. Hum. Fact. Electron.*, 1:4–11. DOI: 10.1109/THFE2.1960.4503259.

J. Light. 1999. When computers were women. *Technol. Cult.*, 40(3): 455–483. Available at http://www.jstor.org/stable/25147356.

S. B. Lipner. 1975. A comment on the confinement problem. SOSP '75 *Symposium on Operating System Principles*, pp. 192–196. DOI: 10.1145/1067629.806537.

S. B. Lipner. 2015. The birth and death of the Orange Book. *IEEE Ann. Hist. Comput.*, 37(2): 19–31. DOI: 10.1109/MAHC.2015.27.

J. S. Long and M. F. Fox. 1995. Scientific careers: Universalism and particularism. *Ann. Rev. Sociol.*, 21:45–71.

B. Longo. 2007. ACM established to develop communication about computing. *Commun. ACM*, 50(5): 27–29. DOI: 10.1145/1230819.1230834.

B. Longo. 2015. *Edmund Berkeley and the Social Responsibility of Computer Professionals.* Association for Computing Machinery and Morgan & Claypool, New York. DOI: 10.1145/2787754.

Iu. M. Lotman and B. A. Uspenskii. 1987. *Pis'ma russkogo puteshestvennika* Karamzina i ikh mesto v razvitii russkoi kul'tury. Introductory note to *Pis'ma russkogo puteshestvennika* by N. M. Karamzin, pp. 525–606. Nauka, Leningrad.

R. S. Lowen. 1997. *Creating the Cold War University: The Transformation of Stanford.* University of California Press, Berkeley.

J. Lucena. 2005. *Defending the Nation: U.S. Policymaking to Create Scientists and Engineers from Sputnik to the 'War against Terrorism'.* University Press of America, Lanham, MD.

R. MacDougall. 2007. All red dream: Technological nationalism at the Trans-Canada Telephone System. In N. Hillmer and A. Chapnick, editors. *Canadas of the Mind: The Making and Unmaking of Canadian Nationalisms in the Twentieth Century*, pp. 47–62. McGill-Queen's University Press, Montreal & Kingston.

D. MacKenzie. 2001. *Mechanizing Proof: Computing, Risk, and Trust.* MIT Press, Cambridge, MA.

M. S. Mahoney. 2001. Boys' toys and women's work: Feminism engages software. In A. N. H. Creager, E. Lunbeck, and L. Schiebinger, editors. *Feminism in Twentieth Century Science, Technology and Medicine*, pp. 169–185. University of Chicago Press, Chicago.

M. Mahoney. 1997. Computer science: The search for a mathematical theory. In J. Krige and D. Pestre, editors. *Science in the Twentieth Century*, pp. 617–634. Harwood Academic Publishers, Amsterdam.

M. Mahoney. 2002. Software: The self-programming machine. In A. Akera and F. Nebeker, editors. *From 0 to 1: An Authoritative History of Modern Computing*, pp. 91–100. Oxford University Press, New York.

M. S. Mahoney. In T. Haigh, editor. 2011. *Histories of Computing.* Harvard University Press, Cambridge, MA.

S. Marenco. 2013. *Barbie I Can Be . . . a Computer Engineer.* Random House, New York.

J. Margolis and A. Fisher. 2002. *Unlocking the Clubhouse: Women in Computing*. MIT Press, Cambridge, MA.

C. J. Martincic and N. Bhatnagar. 2012. Will computer engineer Barbie impact young women's career choices? *Inform. Syst. Educ. J.*, 10(6): 4–14.

J. Matthews. 1989. ACM Forum: Worm turns. *Commun. ACM*, 32(9): 1044.

D. B. Mayer. November 18–20, 1969. The involved generation: Computing people and the disadvantaged. *AFIPS '69 (Fall) Proceedings of the November 18–20, 1969, Fall Joint Computer Conference*, pp. 679–690. ACM, New York. DOI: 10.1145/1478559.1478642.

D. R. McCamus. 1970. They worry too much. *CIPS Mag.*, 1(2): 21.

D. D. McCracken. 1974. A problem-list of issues concerning computers and public policy. *Commun. ACM*, 17(9): 496–503.

D. McCullough. 1988. Noninterference and the composibility of security properties. *IEEE Symposium on Security and Privacy*, Computer Society Press of the IEEE, pp. 177–86.

M. D. McIlroy. 1990. ACM Forum: Green light for bad software. *Commun. ACM*, 33(5): 479.

J. S. McIlwee and J. G. Robinson. 1992. *Women in Engineering: Gender, Power, and Workplace Culture.* SUNY Press, Albany.

A. M. McMahon. 1984. *The Making of a Profession: A Century of Electrical Engineering in America.* Institute of Electrical and Electronics Engineers, New York.

R. Merkle. 1978. Secure communications over insecure channels. *Commun. ACM*, 21(4): 294–299. DOI: 10.1145/359460.359473.

L. Mezei. 1965. The social responsibility of computer specialists. *Computing and Data Processing Society of Canada Quarterly Bulletin*, 5(3): 32–43, 51.

L. Mezei. 1969. Social responsibility. *Information Processing Society of Canada Quarterly Bulletin*, 9(2): 5–6.

J. Minker. 1981. Computer scientists whose scientific freedom and human rights have been violated: A report of the ACM committee on scientific freedom and human rights. *Commun. ACM*, 24(3): 134–139. DOI: 10.1145/358568.358584.

T. J. Misa, editor. 2010. *Gender Codes: Why Women are Leaving Computing*. Wiley IEEE Computer Society, Hoboken, NJ.

T. J. Misa. 2010. Gender codes: Defining the problem. In T. J. Misa, editor. *Gender Codes: Why Women Are Leaving Computing*, pp. 3–23. Wiley-IEEE Computer Society, Hoboken, NJ.

T. J. Misa. 2012 & 2013. Charles Babbage Institute: Glimpses of the founding. *CBI Newslett.*, 34(2) and 35(1). Available at www.cbi.umn.edu/about/nsl/v34n2.pdf#page=4 and www.cbi.umn.edu/about/nsl/v35n1.pdf#page=19.

T. J. Misa. 2015. Computing is history. *Commun. ACM*, 58(10): 35–37. DOI: 10.1145/2814845.

T. J. Misa. 2016. Charles Babbage, Ada Lovelace, and the Bernoulli numbers. In R. Hammerman and A. L. Russell, editors. *Ada's Legacy.* ACM and Morgan & Claypool, New York.

R. Montanelli and S. Mamrak. 1976. The status of women and minorities in academic computer science. *Commun. ACM*, 19(10): 578–581. DOI: 10.1145/360349.360361.

L. B. Moore. 1971. Unhindered. *CIPS Mag.*, 2(9): 12–14.

M. G. Morgan, M. R. Mirabito, and N. J. Down. 1970. Computer jobs through training: A preliminary project report. *AFIPS '70 (Fall): Proceedings of the November 17-19, 1970, Fall Joint Computer Conference*, p. 346. ACM, New York.

M. G. Morgan, N. J. Down, and R. W. Sadler. December 5–7, 1972. Computer jobs through training: A final project report. *1972 Fall Joint Computer Conference. AFIPS Conference Proceedings*, Vol. 41, Part II, pp. 1243–1249. American Federation of Information Processing Societies, Inc., Montvale, NJ. DOI: 10.1145/1480083.148016.

P. M. Morse. June 2–4, 1960. Report on a conference of university computing center directors. *Commun. ACM*, 3(10): 519–521. DOI: 10.1145/367415.993448.

A. L. Morton, Jr. May 14–16, 1969. Computers and the underprivileged. *AFIPS '69 (Spring) Proceedings of the May 14–16, 1969, Spring Joint Computer Conference*, pp. 37–38. ACM, New York. DOI: 10.1145/1476793.1476806.

P. Mounier-Kuhn. 2012. Computer science in French universities: Early entrants and latecomers. *Inform. Cult. J. Hist.*, 47(4): 414–456.

M. C. Mulder. 1975. Model curricula for four-year computer science and engineering programs: Bridging the tar pit. *Computer*, 8(12): 28–33. DOI: 10.1109/C-M.1975.218831.

M. C. Mulder and J. Dalphin. 1984. Computer science program requirements and accreditation. *Commun. ACM*, 27(4): 330–335. DOI: 10.1145/358027.358036.

W. Myers. 1986. Computing sciences accreditation pro and con. *Computer*, 19(6): 47–49. DOI: 10.1109/MC.1986.1663252.

T. Narten and E. H. Spafford. 1989. ACM Forum: Beyond worms. *Commun. ACM*, 32(6): 673–674.

R. A. Nelsen. 2017. Race and computing: The problem of sources, the potential of prosopography, and the lesson of *Ebony* magazine. *IEEE Ann. Hist. Comput.*, (forthcoming 2017). DOI: 10.1109/MAHC.2016.11.

T. Nelson. 1974. *Computer Lib/Dream Machines*. Mindful Press, Sausalito, CA.

T. H. Nelson. 1965. Complex information processing: A file structure for the complex, the changing and the indeterminate. In *Proceedings of the 20th National Conference*, pp. 84–100. ACM Press, New York. DOI: 10.1145/800197.806036.

M. Nerland and B. Karseth. 2015. The knowledge work of professional associations: Approaches to standardisation and forms of legitimisation. *J. Educ. Work*, 28(1): 1–23. DOI: 10.1080/13639080.2013.802833.

J. J. Nersessian. 2010. *Creating Scientific Concepts.* MIT Press, Cambridge, MA.

B. Neugent. 1986. Preposterous opinions about computer security. *SIGSAC Rev.*, 4(3): 1–8. DOI: 10.1145/1058414.1058415.

P. G. Neumann. March 29, 1991. More on Computers, Freedom, and Privacy. Available at http://catless.ncl.ac.uk/Risks/11.35.html#subj3.1

P. G. Neumann, S. Peisert, and M. Schaefer. 2014. The IEEE Symposium on Security and Privacy, in retrospect. *IEEE Secur. Priv.*, 12(3): 15–17.

A. Newell, A. J. Perlis, and H. A. Simon. 1967. Letters: Computer science. *Science*, 157(22): 1373–1374. http://www.jstor.org/stable/1723308.

M. Newell and J. Blinn. 1977. The progression of realism in computer generated images. *ACM '77 Proceedings of the 1977 Annual Conference*, pp. 444–448. ACM, New York.

J. Nielsen. 1995. *Multimedia and Hypertext: The Internet and Beyond*. Morgan Kaufmann.

I. Nikiforova. 2012. The paradox of excellence: Merit and occupational attainments of women in computer science. In *Proceedings from GEXcel Theme 11–12: Gender Paradoxes in Changing Academic and Scientific Organisation(s)*, pp. 49–56. LiU-Tryck, Linköping University Press, Linköping.

D. Nofre, M. Priestley, and G. Alberts. 2014. When technology became language: The origins of the linguistic conception of computer programming, 1950–1960. *Technol. Cult.*, 55(1): 40–75. DOI: 10.1353/tech.2014.0031.

A. Norberg. 2003. Twenty-five years of the Charles Babbage Institute. *CBI Newsletter* 26(1). Available at www.cbi.umn.edu/about/nsl/v26n1graphics.pdf#page=3.

A. L. Norberg and J. E. O'Neill. 1996. *Transforming Computer Technology: Information Processing for the Pentagon, 1962–1986*. Johns Hopkins University Press, Baltimore.

J. November. 2007. George Elmer Forsythe. In *The New Dictionary of Scientific Biography*. Charles Scribner's Sons, New York.

J. November. 2012. *Biomedical Computing: Digitizing Life in the United States.* Johns Hopkins University Press, Baltimore.

A. G. Oettinger. 1966. On ACM's responsibility. *Commun. ACM*, 9(4): 246. DOI: 10.1145/365278.365290.

R. Oldenziel. 2003. Multiple-entry visas: Gender and engineering in the US, 1870–1945. In A. Canel, R. Oldenziel, and K. Zachmann, editors. *Crossing Boundaries, Building Bridges: Comparing the History of Women Engineers, 1870s–1990s*, pp. 11–49. Routledge, New York.

J. D. Oldham. 2009. Examining computers and society 1970–2008. *Comput. Soc.*, 39(2): 34–42. DOI: 10.1145/1621918.1621930.

J. R. Oliver. 1974. Computers and government department: Controversial topics invited. *Comput. Soc.*, 5(4): 2.

R. Osborne. 2011. *Civilization: A New History of the Western World*. Random House.

D. Park. 2015. Social life of PKI: Sociotechnical development of Korean public-key infrastructure. *IEEE Ann. Hist. Comput.*, 37(2): 59–71. DOI: 10.1109/MAHC.2015.22.

D. B. Parker. 1976. Computer abuse perpetrators and vulnerabilities of computer systems. In *AFIPS '76 Proceedings of the June 7–10, 1976, National Computer Conference and Exposition*, pp. 65–73. DOI: 10.1145/1499799.1499810.

D. B. Parker. 1981. The impact on computers on trust in business. *Comput. Soc.*, 11(3): 42. DOI: 10.1145/957859.957865.

D. B. Parker. 1989. ACM Forum: Beyond worms. *Commun. ACM*, 32(6): 673.

Z. Patterson. 2009. From the gun controller to the mandala: The cybernetic cinema of John and James Whitney. *Grey Room*, 36: 36–57. DOI: 10.1162/grey.2009.1.36.36.

Z. Patterson. 2015. *Peripheral Vision: Bell Labs, the SC 4020, and the Origins of Computer Art*. MIT Press, Cambridge, MA.

A. Pearl, M. E. Pollack, E. Riskin, E. Wolf, B. Thomas, and A. Wu. 1990. Becoming a computer scientist: A report by the ACM committee on the status of women in computing science. *Commun. ACM*, 33(11): 47–57. DOI: 10.1145/92755.92757.

A. J. Perlis and K. Samelson. 1958. Preliminary report: International algebraic language. *Commun. ACM*, 1(12): 8–22. DOI: 10.1145/377924.594925.

A. J. Perlis and K. Samelson. 1959. *Soobshchenie ob algoritmicheskom iazyke ALGOL*, translated by A. P. Ershov. Vychislitel'nyi Tsentr AN SSSR, Moscow.

A. J. Perlis. 1963. Computation's development critical to our society. *Commun. ACM*, 6(10): 642. DOI: 10.1145/367651.367695.

A. J. Perlis. April 24, 1964. Report of the Commission of Thoughtful Persons to the ACM Council. *Commun. ACM*, 7(8): 507–508.

R. Perry. 1972. The American challenge: Stand on own feet, Perry tells Canada. *CIPS Mag.*, 3(6): 10–11.

J. L. Peterson. 1975. Notes on a workshop on distributed computing. *Operat. Syst. Rev.*, 13(3): 20.

H. Petroski. 2011. *An Engineer's Alphabet: Gleanings from the Softer Side of a Profession.* Cambridge University Press, New York.

J. R. Pierce. 1967. *Computers in Higher Education: Report of the President's Science Advisory Committee*. US GPO, Washington, DC.

G. M. Pike. November, 1973. ICCP–here's how it works. *CIPS Comput. Mag.*, 11.

R. I. Podlovchenko. 2001. A.A. Lyapunov and A.P. Ershov in the theory of program schemes and the development of its logic concepts. In D. Bjørner, M. Broy, and A. V. Zamulin, editors. *Perspectives of System Informatics: 4th International Andrei Ershov Memorial Conference, PSI 2001, Revised Papers*, pp. 8–23. Springer, Berlin/Heidelberg.

S. V. Pollack, editor. 1982. *Studies in Computer Science*. Mathematical Association of America, Washington, DC.

P. Purdom. 1986. ACM Forum: Reported hacking. *Commun. ACM*, 29(7): 88–89.

R. L. Pyle. 2015. *First in the Field: Breaking Ground in Computer Science at Purdue University.* Purdue University Press, West Lafayette, IN.

A. Ralston and C. L. Meek, editors. 1976. *Encyclopedia of Computer Science.* Petrocelli/Charter, New York.

S. K. Reed and D. K. Branstad, editors. December 10–13, 1974. *Controlled Accessibility Workshop Report: A Report of the NBS/ACM Workshop on Controlled Accessibility, December 10–13, 1972, Santa Fe, California*. Government Printing Office, Washington DC.

J. Reeves. 1984. ACM Forum: Trojan horses and trusty hackers. *Commun. ACM*, 27(11): 1086.

J. B. Reid. 1965. CSC and the metric system. *Computer Society of Canada Quarterly Bulletin*, 6(1): 21–24.

L. Revens. 1987. The first 25 years: ACM 1947–1962. *Commun. ACM*, 30(10): 860–865. (Originally published *Commun. ACM*, 15(7): 485–490). DOI: 10.1145/30408.30418.

T. S. Reynolds. 1983. *75 Years of Progress: A History of the American Institute of Chemical Engineers, 1908–1983.* American Institute of Chemical Engineers, New York.

Y. Richmond. 2004. *Cultural Exchange and the Cold War: Raising the Iron Curtain.* Penn State University Press, University Park.

D. M. Ritchie and K. Thompson. 1974. The UNIX time-sharing system. *Commun. ACM*, 17(7): 365–375. DOI: 10.1145/361011.361061.

E. L. Robertson, B. W. Boehm, H. D. Huskey, A. B. Kamman, and M. R. Lackner. 1976. Computers in developing nations. *Comput. Soc.*, 7(2): 7–9. DOI: 10.1145/958838.958840.

M. Rosen, editor. 2011. *A Little-known Story about a Movement, a Magazine, and the Computer's Arrival in Art: New Tendencies and Bit International, 1961–1973.* MIT Press, Cambridge, MA.

D. Rosenberg. 2015. History debugged. In D. R. Dechow and D. C. Struppa, editors. *Intertwingled: The Work and Influence of Ted Nelson*, pp. 91–103. Springer, Heidelberg. DOI: 10.1007/978-3-319-16925-5_13.

M. W. Rossiter. 1993. The ~~Matthew~~ Matilda effect in science. *Soc. Stud. Sci.*, 23(2): 325–341. DOI: 10.1177/030631293023002004.

M. W. Rossiter. 1984. *Women Scientists in America: Struggles and Strategies to 1940.* Johns Hopkins University Press, Baltimore.

M. W. Rossiter. 1995. *Women Scientists in America: Before Affirmative Action, 1940–1972*. Johns Hopkins University Press, Baltimore.

M. W. Rossiter. 2012. *Women Scientists in America: Forging a New World since 1972*. Johns Hopkins University Press, Baltimore.

B. Rous, B. Shneiderman, N. Yankelovich, and E. Yoder. 1989. Lessons learned from the ACM Hypertext on Hypertext Project. In *Proceedings of the Second Annual ACM Conference on Hypertext*, (HYPERTEXT '89), pp. 385–386. ACM, New York. DOI: 10.1145/74224.74255.

M. Rubinoff. 1964. Impact of computers on society. *Commun. ACM*, 7(10): 637–638.

A. L. Russell. 2014. Accreditation and the boundaries of computer science, 1984–1999. Business History Conference, Frankfurt am Main, Germany. Abstract at http://www.thebhc.org/node/2921.

J. E. Sammet. 1966. The use of English as a programming language. *Commun. ACM*, 9(3): 228–230. DOI: 10.1145/365230.365274.

P. J. Sandiford. 1963. The origin and growth of the Canadian Operational Research Society. *Journal of the Canadian Operational Research Society*, 1(1): 1–13.

N. Sawhney, D. Balcom, and I. Smith. 1996. HyperCafe: Narrative and aesthetic properties of hypervideo. In *Proceedings of the Hypertext '96 conference*, ACM Press, Washington, DC.

A. Saxenian. 1994. *Regional Advantage: Culture and Competition in Silicon Valley and Route 128*. Harvard University Press, Cambridge, MA.

M. Schep and N. McNulty. Experiences with using robots in an all-female programming class. ITISCE '02. DOI: 10.1145/637610.544510.

C. Schlombs. 2010. A gendered job carousel: Employment effects of computer automation. In T. J. Misa, editor. *Gender Codes: Why Women are Leaving Computing*, pp. 75–94. Wiley IEEE Computer Society, Hoboken, NJ.

M. Schroeder. 1972. Discussion of 'A Hardware Architecture for Implementing Protection Rings'. *Operat. Syst. Rev.*, 6(1): 166.

R. Schwarz. 1989. On regulation and legislation. *Commun. ACM*, 32(8): 918.

E. J. Schweppe. 1990. On the genesis of Curriculum 68. In *ACM Annual Conference*, Washington, DC.

D. E. Shalala, chair. 2007. The Committee on Maximizing the Potential of Women in Academic Science and Engineering. *Beyond Bias and Barriers: Fulfilling the Potential of Women in Academic Science and Engineering*. National Academies Press, Washington, DC.

R. S. Sharma. 1986. Data communications and security. *SIGSAC Rev.*, 4(1): 28–38.

I. P. Sharp. 1971. Accreditation?–No. *CIPS Mag.*, 2(9): 4–14.

H. A. Simon. 1969. *The Sciences of the Artificial*. MIT Press, Cambridge, MA.

B. Sinclair. 1980. *A Centennial History of the American Society of Mechanical Engineers, 1880–1980*. University of Toronto Press, Toronto.

T. Sito. 2013. *Moving Innovation: A History of Computer Animation*. MIT Press, Cambridge, MA.

R. Slayton. 2015. Measuring risk: Computer security metrics, automation, and learning. *IEEE Ann. Hist. Comput.*, 37(2): 32–45. DOI: 10.1109/MAHC.2015.30.

R. Slayton. 2013. *Arguments That Count: Physics, Computing, and Missile Defense, 1949–2012*. MIT Press, Cambridge, MA.

A. R. Smith. 2001. Digital paint systems: An anecdotal and historical overview. *IEEE Ann. Hist. Comput.*, 23(2): 4–30.

J. B. Smith and F. G. Halasz. 1987. *Proceedings of the ACM Conference on Hypertext*, ACM Press, New York.

J. B. Smith and S. F. Weiss. 1988. Hypertext. *Commun. ACM*, 31(7): 816–819. DOI: 10.1145/48511.48512.

G. Sonnert and G. Holton. 1995. *Who Succeeds in Science?* Rutgers University Press, New Brunswick, NJ.

E. H. Spafford. 1989. Crisis and aftermath. *Commun. ACM*, 32(6): 678–687.

G. Spafford. 2014. We are out of balance. *ACM SIGCAS Comput. Soc.*, 44(4): 9–12. DOI: 10.1145/2695577.2695579.

R. E. Sprague. 1974. Issues for SIGCAS. *Comput. Soc.*, 5(3): 3–4.

Z. Stachniak and S. Campbell. 2009. *Computing in Canada: Building a Digital Future.* Canada Science and Technology Museum, Ottawa.

Z. Stachniak. 2011. *Inventing the PC.* McGill-Queen's University Press, Montreal.

Richard Stallman. 1984. Hacking away at morality. *Commun. ACM*, 27(1): 8.

B. Sterling. 1992. *The Hacker Crackdown: Law and Disorder on the Electronic Frontier.* Available at www.gutenberg.org/ebooks/101.

P. Stewart. May 1968. Faculty profile: David C. Evans. *Utechnic: University of Utah Engineering Publication*, p. 10.

J. Stout and T. Camp. 2014. Now what? Action items from social science research to bridge the gender gap in computing research. *ACM SIGCAS Comput. Soc.*, 44(4): 5–8. DOI: 10.1145/2695577.2695578.

M. Strathern. 2004. *Commons and Borderlands: Working Papers on Interdisciplinarity, Accountability and the Flow of Knowledge*. Sean Kingston Publishing, Oxford, UK.

R. H. Stuewer. editor. 1970. *Minnesota Studies in the Philosophy of Science*, vol. 5, pp. 112–127. University of Minnesota Press, Minneapolis.

R. Subramanian and T. Haigh. 2013. Starr Roxanne Hiltz: Pioneer digital sociologist. *IEEE Ann. Hist. Comput.*, 35(1): 78–95.

I. Sutherland. 1965. The ultimate display. *Proceedings of IFIP Congress*, pp. 506–508. DOI: 10.1.1.136.3720.

I. Sutherland. 1966. Computer graphics: Ten unsolved problems. *Datamation*, 12(5): 22–27.

K. Tatarchenko. 2010. Cold War origins of the International Federation for Information Processing. *IEEE Ann. Hist. Comput.*, 32(2): 46–57. DOI: 10.1109/MAHC.2010.8.

K. Tatarchenko. 2011. Informatika ot Silikonovoi Doliny do Zolotoi Doliny. In A. N. Tomilin, editor. *SoRuCom–2011 Proceeding, Second International Conference on the History of Computers and Informatics in the Soviet Union and Russian Federation*, Veliky Novgorod, pp. 278–282.

K. Tatarchenko. 2013. 'A house with the window to the West': The Akademgorodok Computer Center, 1958–1993. Ph.D. dissertation, Princeton University.

O. E. Taulbee and S. D. Conte. 1977. Production and employment of Ph.D.'s in computer science—1976. *Commun. ACM*, 20(6): 370–372. DOI: 10.1145/359605.359608.

M. Tedre. 2014. *The Science of Computing: Shaping a Discipline*. Chapman and Hall/CRC, London/New York.

A. Thackray. 1970. Science: Has its present past a future? In R. H. Stuewer, editor. *Historical and Philosophical Perspectives of Science*, pp. 112–132. University of Minnesota Press, Minneapolis.

L. Thurgood, M. J. Golladay, and S. T. Hill. 2006. *U.S. Doctorates in the 20th Century*. National Science Foundation, Arlington, VA.

J. Titus. 1967. Security and privacy. *Commun. ACM*, 10(6): 379–380.

J. P. Titus. 1967. ARPA: A visible means of support (Washington Commentary). *Commun. ACM*, 10(8): 519–520. DOI: 10.1145/363534.363564.

J. P. Titus. 1967. The patchwork nature of government computing support (Washington Commentary). *Commun. ACM*, 10(9): 589–592. DOI: 10.1145/363566.363703.

A. B. Tucker and N. E. Gibbs. 1984. On accreditation. *Commun. ACM*, 27(5): 411–412. DOI: 10.1145/358189.381145.

A. B. Tucker, editor. 1991. Computing curricula 1991. *Commun. ACM*, 34(6): 68–84. DOI: 10.1145/103701.103710.

R. Turn, E. J. Novotny, J. J. Geraghty, E. H. Sibley, and W. H. Ware. 1982. *Observations on the Resilience of the U.S. Information Society*. American Federation of Information Processing Societies, Arlington, VA.

R. Turn. 1975. Security problems in computer communication systems. *Comput. Commun. Rev.*, 5(1): 35–44. DOI: 10.1145/1024847.1024851.

L. Turner. November, 1971. Canadian DPMA Institute formed to deal with domestic problems. *Canadian Datasystems*, pp. 36–37, 49.

L. Turner. 1970. Professionalism: Curse or course. *CIPS Mag.*, 1(4): 21.

A. Van Dam. 1988. Hypertext '87 Keynote Address. *Commun. ACM*, 31(7): 887–895. DOI: 10.1145/48511.48519.

J. N. Vardalas. 2001. *The Computer Revolution in Canada.* MIT Press, Cambridge, MA.

E. J. Vettel. 2004. The protean nature of Stanford University's biological sciences 1946–1972. *Hist. Stud. Phys. Biol. Sci.*, 35(1): 95–113.

D. Walden and T. Van Vleck, editors. 2011. *The Compatible Time Sharing System (1961–1973): Fiftieth Anniversary Commemorative Overview*. IEEE Computer Society.

H. M. Walker and C. Kelemen. 2010. Computer science and the liberal arts: A philosophical examination. *ACM Trans. Comput. Educ.*, 10(1): 2:1–2:10. DOI: 10.1145/1731041.1731043.

W. H. Ware. 1959. Soviet computer technology: 1959. RAND Corporation, Santa Monica, RM-2541. DOI: 10.1145/367149.1047530.

W. H. Ware. 1960. Soviet computer technology—1959. *Commun. ACM*, 3(3): 131–166. DOI: 10.1145/958858.958859.

W. H. Ware. 1967. Security and privacy in computer systems. In *Proceedings of the April 18–20, 1967, Spring Joint Computer Conference AFIPS '67*, pp. 279–280. ACM Press, New York.

W. H. Ware. 1977. Testimony before the National Commission of Electronic Fund Transfer. *Comput. Soc.*, 8(1): 2–3.

W. H. Ware. 1977. Impact of telecommunications technology on the right to privacy. *Comput. Soc.*, 8(4): 2–5. DOI: 10.1145/958901.958902.

W. H. Ware. 1984. Information systems security and privacy. *Commun. ACM*, 27(4): 315–321.

W. H. Ware. 2008. *RAND and the Information Evolution: A History in Essays and Vignettes*. RAND Corporation, Santa Monica.

M. Warner. 2015. Notes on the evolution of computer security policy in the US government, 1965–2003. *IEEE Ann. Hist. Comput.*, 37(2): 8–18. DOI: 10.1109/MAHC.2015.25.

J. W. Weaver. 1989. ACM Forum: More on regulation. *Commun. ACM*, 32(11): 1284.

R. Weber and B. Gilchrist. 1975. Discrimination in the employment of women in the computer industry. *Commun. ACM*, 18(7): 416–418. DOI: 10.1145/360881.360921.

D. Weikle. 2014. Letter from the Editor. *ACM SIGCAS Comput. Soc.*, 44(4): 3.

F. Weingarten. 1981. Cryptographic research and the computer science community. *Commun. ACM*, 24(12): 851–853.

M. Wetherell, S. Taylor, and S. J. Yates. 2001. *Discourse as Data: A Guide for Analysis*. Sage Publications, London.

P. Wettersten. 1986. ACM Forum: Further observations on hacking. *Commun. ACM*, 29(9): 829.

E. Weyuker. 2009. ACM-W celebrates women in computing. *Commun. ACM*, 52(6): 5. DOI: 10.1145/1516046.1516047.

B. White. 2013. SIGWEB History Project. *ACM SIGWEB Newslett.*, (Spring 2013): 1–5. DOI: 10.1145/2451836.2451837.

S. B. Williams. 1954. The Association for Computing Machinery. *J. ACM*, 1(1): 1–3. DOI: 10.1145/320764.320765.

J. A. Wills. 1986. ACM Forum: Reported hacking. *Commun. ACM*, 29(7): 590.

J. M. Wing. 2006. Computational thinking. *Commun. ACM*, 49(3): 33–35. DOI: 10.1145/1118178.1118215.

M. Wisnioski. 2012. *Engineers for Change: Competing Visions of Technology in 1960s America.* MIT Press, Cambridge, MA.

A. Wright. May 2014. The secret history of hypertext. *The Atlantic*. Available at www.theatlantic.com/technology/archive/2014/05/in-search-of-the-proto-memex/371385/.

R. Wright and J. A. Jacobs. 1994. Male flight from computer work: A new look at occupational resegregation and ghettoization. *Am. Sociolog. Rev.*, 59(4): 511–536.

M. Wyer, M. Barbercheck, D. Cookmeyer, H. Ozturk, and M. Wayne, editors. 2013. *Women, Science, and Technology: A Reader in Feminist Science Studies.* Routledge, New York.

J. R. Yost. 2007. A History of computer security standards. In K. De Leeuw and J. A. Bergstra, editors. *The History of Information Security: A Comprehensive Handbook*, pp. 595–621. Elsevier, Amsterdam.

J. R. Yost. 2008. From the Editor's desk. *IEEE Ann. Hist. Comput.*, 30(1): 2–3.

J. R. Yost. 2015. The origin and early history of the computer security software products industry. *IEEE Ann. Hist. Comput.*, 37(2): 46–58. DOI: 10.1109/MAHC.2015.21.

M. Young and J. Muller. 2014. *Knowledge, Expertise and the Professions*. Routledge, London/New York.

Z. Zeller. 1987. *Inventing Canada.* University of Toronto Press, Toronto.

H. Zuckerman, J. R. Cole, and J. T. Bruer, editors. 1992. *The Outer Circle: Women in the Scientific Community*. Yale University Press, New Haven, CT.

L. M. Zurk, et al. 1995. When an advantage is not an advantage. *Commun. ACM*, 38(12): 17–18.

Index

Abbate, Janet, 13–15
  Recoding Gender, 77
  women's contributions to computing, 146–147
"Academy of Death and Destruction", 219
Accreditation
  ACM role, 96–101
  CIPS, 190–195
  design in, 104
  guidelines, 105
  organizational interaction, 108
  resistance to, 197–198
Accreditation Board for Engineering and Technology (ABET), 17, 98–101
Accreditation Committee, 247
ACM Data-link Bulletin, 219
Ada Lovelace Award, 153
Ada Lovelace Day, 153n
Ada Project, 155
Adams, J. M., 256
Adams, John, 128
Adelman, Leonard, 309
Adrion, W. R., 32n
Advanced Research Projects Agency
  Center of Excellence, 271
  Information Processing Techniques Office, 260, 264, 269
Advisors for doctoral women in computer science, 81
Advisory Committee on Labor Management Policy, 234
Advisory Panel on University Computing Facilities, 31
Affirmative-action resentment, 164–165
African-Americans. *See* Training and education for communities of color
Aiken, Howard, 334
Akademgorodok Computer Center, 202–203, 213–214, 216–217, 222–224
Akeley, Kurt, Papers, 262n
Akera, Atsushi, 209
Akscyn, Robert, 340–341, 343–344
Algol, 10, 206–209, 212
Algol Bulletin, 207, 209
Algol group, 206–208
Allen, Frances, 23, 86
Allen Babcock Computing, 204
Amdahl, Gene, 215
American Association for Cybernetics, 211
American Association of University Women (AAUW), 157, 165–166
American Chemical Society (ACS)
  accreditation model, 98
  history center, 11
  Women's Service Committee, 151
American Council on Education, 318
American Federation of Information Processing Societies (AFIPS)
  conferences, 5
  Fall Joint Computer Conference, 234–235, 239, 241

American Federation of Information Processing Societies (AFIPS) *(continued)*
- privacy issues, 123–124
- security issues, 289
- security issues study, 314–315

American Institute of Electrical Engineers (AIEE), 2, 5, 26
American Institute of Physics, 11
American Mathematical Society, 32–33
American Physical Society, 151
American Society of Civil Engineers (ASCE), 150
Amnesty International, 119
Anderson, Ronald, 133
Angluin, Dana, 78
- Computational Learning Theory Conference, 85
- places of employment, affiliations, and research areas, 84
- university, dissertation, and first position, 80

Anita Borg Institute for Women Technology
- advocacy by, 5–6
- Computer Science Education Week, 158

Annals of the History of Computing, 5, 7, 10
Annual Computer Security Applications Conference, 328
Annual Conference on the Theory and Practice of Digital Libraries, 343–344
Anti-Ballistic Missiles, 114–115, 122n
Application of Computers to the Problems of Urban Society symposium, 249
Arguments that Count, 112
Armer, Paul
- CCPP member, 122n
- Commission of Thoughtful Persons, 35n
- Committee on Government Relations, 36
- Committee on Social Responsibilities chair, 233n
- Computer Professionals against Anti-Ballistic Missiles, 114
- Ershov visit, 200–201, 219
- international relationships, 222–224
- privacy issues, 296
- security research, 289
- social issues, 126

Armer collection, 289
Armstrong, Louis, 222
Art of Computer Programming, 14, 48
"As We May Think" article, 337
Ashenhurst, Robert, 122n, 125n, 129
Ashton, Alan, 281
Aspray, William
- Adrion interview, 32n
- computer science consolidation, 202
- on gender identity, 148
- international computing study, 209
- on Rosser Report, 41n
- on von Neumann grant, 31n
- Weingarten interview, 27n

Association for Computational Learning (ACL), 85
Association for Symbolic Logic, 33
Association for Women in Computing (AWC), 5, 18, 144
- vs. ACM-W, 152–153
- Career Development Guide, 156
- Equal Rights Amendment, 168
- establishment, 154
- female advocacy, 167–168
- hospitality suite sponsorship, 145
- K-12 outreach for women, 158
- logo, 153

Association for Women in Mathematics, 151
Association for Women in Psychology, 151
Association of Information Technology Professionals (AITP), 5
Associative linking, 337
Atchison, William F., 94n, 103n
Atoms for Peace program, 205–206
Atta, Bruce Van, 115, 122n
Austing, Richard, 132
Authentication, 308
Automation and unemployment, 3, 72n, 182, 233–235, 248n

Bachman, Charlie
ACM archives, 11
Turing laureate, 23
Backus, John, 7, 23
Bajcsy, Ruzena, 78
places of employment, affiliations, and research areas, 84
university, dissertation, and first position, 80
Ball, Leslie, 309
Banta, Darlene, 171
Barber, Donald T., 186
Barbie doll, 159–162
Bardini, Thierry, 269
Bariff, Martin, 309
Barker, Lecia, 148
Barr, Valerie, 169
Barton, Robert, 278
Bauman, Milton, 245, 255
Belanger, Dian Olson, 28n, 44n
Belford, Pauline, 170
Bell, Gwen, 73
Bendix Corporation, 264
Bendix G-15 computer, 264–265
Bendix G-20 computer, 264
Bergin, Tim, 6–7
Berkeley, Edmund C.
ACM archives, 11
ACM founding, 2–3
ACM purpose, 26
Committee on Computing and the Disadvantaged, 232–233
Silver Banquet Committee address, 131
social activism, 113–115, 232
social responsibility of computing, 155n
Use of Computers to the Benefit of Mankind committee, 125
Berners-Lee, Tim, 22, 342–343
Bernstein, Mark, 341
Berson, Tom, 309–310
BESM-6 computer, 223
“Big Dream” film, 158
Bigelow, Robert P., 126–129, 133, 298
Bilas, Frances, 144
Bilingual instruction, 245
Bix, Amy Sue, 18
Blatch, Nora Stanton, 150
Blinn, Jim, 276n, 277
Bloombecker, Jay “Buck”
computer crime, 299
hacker debate, 320
Bluegrass Community College, 158
Boggs, J. Caleb, 233
Boise State University, 158
Bolter, J. D., 341
Borg, Anita
Institute for Women and Technology, 152
“Minerva’s Machine: Women and Computing” documentary, 155
Boundary-work theory, 28
Bowen, H. Kent, 259n
Bowker, Albert, 56
Bowker, Geof, 211
“Boys’ Toys and Women’s Work: Feminism Engages Software”, 67–68
Brandin, David
cryptography issues, 318
vulnerabilities study, 314
Brezhnev, Leonid, 121
Bright, Herbert S., 122n, 255
Brooks, Fred
HOPL conferences, 7
Turing laureate, 23
Brooks, Harvey, 26n, 34–35
Brown, Cecil Page, 246
“Browsing the WWW by interacting with a textual virtual environment” paper, 343
Burrows, James, 253
Bush, Vannevar, 22, 337
Bushnell, Nolan, 21, 281
Business Data Processing (SIGBDP), 307

Cage, Matilda J., 75–76n
Cain, Tom, 101
Caldwell, Samuel H., 333
Califano, Joseph, 43n
Caltech, 204, 266n

Camp, Tracy, 164, 170
Campbell, Scott, 18
Canada, computer professionalization in
  conclusion, 195–198
  data processing societies, 179–184
  DPMA, 184–194
  early history, 174–176
  early organizations, 176–178
  introduction, 173–174
Canadian Broadcasting Corporation (CBC), 196
Canadian Computer Show and Seminar, 197
Canadian Data Processing and Management Association Institute (CDPMAI), 192
  accreditation, 193–194
  founding, 187
  logo, 197
  purpose, 193
Canadian Datasystems, 189, 196
Canadian Federation of Information Societies (CFIS), 190
Canadian Information Processing Society (CIPS), 5, 19, 176
  accreditation, 190–194
  membership growth, 183
Canadian Journal of Physics, 178
Canadian Operations Research Society (CORS), 184
"Canadian Problem", 19, 187–189
Careers in computing for doctoral women in computer science, 79–83
Carlson, Walter M.
  accreditation, 97–98
  Bauman appointment, 255
Carnegie Institute of Technology, 14, 32, 38, 58
Carr, John W., 206–207, 209
Carr, John W., III, 35n
Carr, Stephen, 280–281
Catmull, Ed, 21, 283
Center of Excellence, 271–272, 283
Central Data Processing Service Bureau, 190
Cerf, Vinton, 23
Certificate in Data Processing (CDP), 189–190
Charland, Maurice, 196
Charles Babbage Institute (CBI)
  AFIPS support, 5
  hosting proposals, 10–11
  Utah history, 262
China Computer Federation (CCF), 92
Chow, Kao Liang, 52
CIA, 211
CIPS Magazine, 184, 193
Civil rights, 239–240
Civil rights. *See also* Training and education for communities of color
Civil Rights Act, 76
Clark, Dave, 313
Clarke, Lori, 78
  ACM-SIG on programming languages and software engineering, 85
  places of employment, affiliations, and research areas, 84
  university, dissertation, and first position, 80
Classification problem, 102
Clipper Chip, 328
Closed World, 226
Codd, Edgar, 23
Code of Fair Information Practices, 295
Cohoon, Joanne McGrath, 148
Colby, Kenneth, 215
Cold war computer science, 19
  conclusion, 226–227
  Ershov trip itinerary, 203–210
  Ershov trip overview, 199–203
  hazards, 218–226
  international cooperation, 210–218
Columbia University
  ACM discussed at, 2
  Computing Center courses, 252
Commission of Thoughtful Persons, 35, 38
Committee of Concerned Scientists, 122
Committee on Computers and Public Policy (CCPP), 114
  issues list, 135

overview, 122–125
SIGCAS relationship, 132
system reliability and risks, 316
"Committee on Computing & Disadvantaged Receives Grant" article, 254
Committee on Computing and the Disadvantaged, 20, 231n
ACM appointment, 233
chair resignation, 255
conference discussions, 254
educational programs, 248
formation, 230, 252
grant usage proposals, 256
mentioned by Galler, 237
termination, 256–257
Committee on Computing Education in Liberal Arts Colleges, 107
Committee on Government Relations, 35–36
Committee on Science and Public Policy, 33
Committee on Scientific Freedom and Human Rights, 121–122, 155
Committee on Support of Research in the Mathematical Sciences (COSRIMS), 33–34
Committee on the Professional Activities of the Blind, 237
Committee on the Status of Women in Computer Science, 155, 163–164
Committee on the Status of Women in Physics, 151
Committee on Women in Computing (ACM-W), 18, 144
alliances, 155
vs. AWC, 152–153
feminist advocacy, 164–165, 167
K-12 outreach for women, 157–158, 160
on meritocracy, 169
Committee to Assess the Scope and Direction of Computer Science and Technology, 164
Communications of the ACM
ACM history, 23, 231
Algol report, 207
Canadian contributions, 178
Canadian leadership, 19
Clipper Chip, 328
communities of color, 230–233, 248–252
computer science degrees, 105
confinement problem, 303
description, 2
"Discrimination in the Employment of Women in the Computer Industry", 76
hypertext, 346
international community, 217
Morris worm, 323–324
Neidorf case, 326–327
programming languages, 7
security and privacy articles, 289–291
social issues, 20, 132–133
vision for academic computing, 58–60
women under-representation in degrees, 163
COMPCON meeting, 154–155
Computation Center, 52–53
Computation Centre, 176–178
Computational Learning Theory (COLT) Conference, 85
Computer and Information Science and Engineering (CISE), 14, 27
"Computer and Information Sciences: A New Basic Discipline", 37
"Computer and the Ghetto: a Means of Social Change" lectures, 249
"Computer boys", 146
Computer Boys Take Over, 247
Computer Center, 273
Computer centers, 94
Computer Engineer Barbie, 159–162
Computer Fraud and Abuse Act, 320, 323–324, 326
"Computer Graphics: Ten Unsolved Problems" article, 274–276
Computer History Museum, 12, 262n
Computer Industry Martin Luther King Fund, 237

Computer Jobs Through Training program, 243–245, 247–248
Computer Museum in Boston, 12
Computer Newsline, 219
Computer Personnel Development Association (CPDA)
civil rights, 240
Computing Center courses, 252
training program, 243–245
Computer Professionals against Anti-Ballistic Missiles, 114–115
Computer Professionals for Peace (CPP), 126–128
civil rights, 239–240
tension with ACM, 238–239
Computer Professionals for Social Responsibility (CPSR)
founders, 114
nuclear war issues, 315
"Computer Revolution: Its Effect on U.S. Business and Labor" panel, 248
Computer school fraud, 127, 249
Computer science
accreditation issues. *See* Accreditation
computing curricula. *See* Computing curricula
definitions, 14, 58–69
discipline issues, 28–30, 101–108
doctoral women. *See* Doctoral women in computer science
early degrees, 26, 97
Forsythe early efforts, 47–51
funding issues, 25–28
Harvey Mudd College, 167
NFS grants, 31–32
roadblocks to, 51–56
scientific identity for. *See* Scientific identity for computer science
Stanford early program, 56–58
subject areas, 103
women degrees, 146
women jobs, 163
Computer Science: A First Course, 69
Computer Science Accreditation Commission (CSAC), 100
Computer Science and Computer Operations, 266–267
Computer Science Association (CSA), 183
Computer Science Department (Stanford)
founded by Forsythe, 15, 48
marginalization of women, 67–68
push for, 56–58
success of, 51–52
Computer Science Education Week, 158
Computer Science Teachers Association, 158
Computer Sciences Man-of-the-Year award, 76
Computer Security Act, 320
Computer Society of Canada, 182–183
Computer Usage company, 204
"Computers '64: Problem-Solving in a Changing World" conference, 234
Computers and Automation, 114, 232, 238
Computers and communications, 236
Computers and Society newsletter
controversial material, 130–131
first issue, 128
SIGCAS, 113, 129–130
subject categories, 138
Computers, Freedom, and Privacy conferences, 21, 312
Computers in Higher Education, 41
Computerworld, 77
Computing and Data Processing Society of Canada (CDPSC), 19, 176, 179–184
Computing as a Discipline report, 103–104
Computing curricula
accreditation, 96–101
conclusion, 108–109
discipline definition, 101–108
early efforts, 91–92
volunteer efforts, 92–96
Computing Educators Oral History Project, 23
Computing Research Association, 5
Committee on the Status of Women, 152

Computer Science Education Week, 158
Computing Sciences Accreditation Board (CSAB), 101
Concentrated Employment Program, 246
Conference Board of Mathematical Sciences, 32–33
Conference on Computers, Freedom, and Privacy, 326–327
Conference on Electronic Digital Computers and Information Processing, 204–205
Conference on Information and Knowledge Management (CIKM), 348
Conference on Web Search and Data Mining (WSDOM), 348
Confinement problem, 303
Control Data Corporation, 264
Conversational Use of Computers (CONCOMP) project, 271
Corbató, Fernando, 23, 179
CORS Journal, 184
"Council Decides Against Support in Clark Squire Case" article, 240
Counter culture, 111
Cray, Seymour, 12
Cray 2 supercomputer, 5
Critical infrastructure risks, 313–316
Cryptography, 288, 307–309
Cultural Exchange Program, 222
Cultural factors and doctoral women, 72
Current Population Survey, 147
Curriculum '68
  release of, 95
  reputation concerns, 194n
Curriculum '68 report, 14, 102
Curriculum Development Subcommittee (CDS), 94–95
Curtis, Kent, 31, 43–44n
Curtiss, John, 334
"Cybernated Generation" article, 59
Cybernetic Council of the Soviet Academy of Sciences, 212
Cybernetics, 211–212
Dalphin, John, 99
Darmstadt conference, 209
Data Processing Management Association (DPMA)
  Canada, 174, 176, 184–194
  Computer Sciences Man-of-the-Year award, 76
  description, 5
Data processing societies in Canada, 179–184
Datamation
  AFIPS conference, 239
  "Computer Graphics: Ten Unsolved Problems" article, 274–275
  Feigenbaum trip to Soviet Union, 222–223
  survey of women, 76–77
D'Auria, Thomas, 119–121
David C. Evans Collection, 262
Davida, George
  encryption papers, 307–308
  encryption patent, 318
  privacy symposium, 311
Davidson, Charles, 128
Davis, Al, 280
Dawson, Helaine S., 254–255
Dayton conference, 154
Defense Department, 305–306
Defense Science Board study, 292
Definitions
  computer science, 14, 58–59
  discipline, 101–108
DeLegall, Walter A.
  Computing Center courses, 252
  cynicism about training motivations, 245–246
  programs for disadvantaged, 254
  training for disadvantaged, 242–243
Democratic National Convention of 1968, 116
Denning, Dorothy
  doctoral dissertation, 302–303
  information control, 307
  Neidorf case, 326–327
  Symposium on Security and Privacy, 312

Denning, Peter, 17
accreditation, 98
Computing as a Discipline report, 103
computing curricula, 95–96, 109
“design” term, 104
Equal Rights Amendment, 118
on McCracken, 115
Morris worm, 324–325
security issues, 313
social activism, 113
on social issues, 116–118
Turchin issue, 119–121
Dennis, Jack, 300–301
Dexter group, 22
Dexter Hypertext Reference Model, 339–340
Diffie, Whitfield
cryptography, 317
Turing laureate, 23
Digital Computer Needs in Universities and Colleges report, 40
Digital Equipment Corporation, 12
Digital libraries, 344
Dijkstra, Edsger
on computer science, 14
on early computer departments, 73
Turing laureate, 23
Disadvantaged. *See* Training and education for communities of color
Discipline
in cold war computer science. *See* Cold war computer science
computer science as, 28–30
definition, 101–108
Discrimination concerns, 127. *See also* Training and education for communities of color
“Discrimination in the Employment of Women in the Computer Industry”, 76
Discursive boundary-work, 36–42
Dissertations and Theses database, 77–78
Division of Computer Science, 15, 51
Division of Mathematical Sciences
National Academy of Sciences, 33
Purdue, 66
Dobrynin, Anatoly, 121
Doctoral women in computer science
advisors, 81
career choice findings, 79–81
careers in computing, 82–83
changes in computing over time, 83–85
conclusion, 89–90
data and method of study findings, 77–79
degrees, 163
family issues, 87–89
first jobs, 81–82
gender and computing, 76–77
gender and science, 74–75
gender issues, 87
graduate schools findings, 79
historical context, 73–74
introduction, 71–73
places of employment, affiliations, and research areas, 84
recognition, 86
scientific communities, 85–86
Document Examiner system, 338
Dolls, 159–162
Dorodnitsyn, A. A.
Ershov letter, 225
IFIP and Algol activities, 208
Soviet computer center, 205
Dorsey, W. Smith, 112
“Double loyalties”, 220–221
DPMA Magazine, 188
Dupree, Andrea, 75
Dziallas, Sebastian, 16

E911 file, 326–327
Eastern Association for Computing Machinery, 2, 333
Eaves, Elsie, 150
Ebony magazine, 242
Eckert, J. Presper, 144
Ecole Polytechnique, 281
“Economics of Programming” panel, 247
Edmund Berkeley and the Social Responsibility of Computer Professionals, 23

Education. *See* Computer science; Training and education for communities of color
Education Amendments of 1972, 76
Education Board
accreditation, 97–98, 105
computing curricula, 92–96
Education Committee, 94
"Education of Non-Computer People" panel, 250
Educational Activities Board, 99–100
EDVAC, 1
Edwards, Paul, 226
Electronic funds transfer systems, 293
Electronic News, 223–225
Elkind, Edward, 128, 240
Elliott, R. C., 187–188, 193
"Employing the Unemployable" article, 246
Employment and automation, 233–235
Encryption
groups interested in, 307–308
privacy issues, 317–318
Engel, Gerald, 103, 132
Engelbart, Douglas, 22
hypertext, 337
Turing laureate, 23
Engelbart Best Paper Award, 343
Engineering sciences, 27–28n
Engineers, women trained as, 73
Engineers' Council for Professional Development, 98–99n
"ENIAC girls", 144
ENIAC machine, 144
Ensmenger, Nathan
accreditation, 189
Computer Boys Take Over, 247
computer science legitimacy, 202
on women as programmers, 146
on women pushed out as programmers, 67
Environmental factors for women in science, 165
Equal Pay Act, 76
Equal Rights Amendment, 116–118, 168
Ershov, Andrei, 19, 199–227
Estrin, Thelma
as engineer, 73
"Minerva's Machine: Women and Computing" documentary, 155
Ethics, 236
Evans, David C., 20–21
Center of Excellence, 271–272
commercial graphics, 283–284
community building, 280–281
Computer Center, 273–274
focus on graphics, 272–273
Mansfield Amendments, 282–283
move to Salt Lake City, 277–279
Utah graphics, 260, 262–264, 266–267
Evans, Joy, 278
Evans & Sutherland Computer Corporation (E&S), 262, 278, 283
"Everyday sexism", 170
Executive Committee
DPMA, 186–187
political issues, 112
SIGCAS controlled by, 128–129
social activism issue, 117
Turchin issue, 120
External Activities Board, 255–256

Faculty positions, women in, 163
Family issues for doctoral women, 87–89
"Father of Silicon Valley", 57
Fawkes, George, 187
Feigenbaum, Edward, 15, 19
career change, 212
Ershov relationship, 213–216, 222
international encounters, 221–222
recruited by Forsythe, 58
Sisakian letter, 221
Siscoe letter, 221
"Soviet Computer Science, Revisited", 212–213
on Soviet Union, 199
Turing laureate, 23
Fein, Louis
computer science study, 52–54

Fein, Louis *(continued)*
 computer science vision, 60
 discipline criteria, 37
Feingold, Sam, 251
Feldman, Jerome A., 122n
Feldman, Sydney, 128, 130
Ferranti computers, 176–178
Ferranti-Packard Canada, 174
Fetter, William, 284
Fierheller, George, 191
Fiesler, Casey, 161
Fincher, Sally, 16
Finite-Difference Methods for Partial Differential Equations, 51
First jobs for doctoral women, 81–82
Fletcher, James, 262, 266–267, 278
FORMAC, 7
Forsythe, Alexandra, 49
 Computer Science: A First Course, 69
 marginalization of women, 68
 SWAC construction followed by, 50
Forsythe, Diana E.
 on father's writings, 49
 marginalization of women, 68
Forsythe, George E., 13, 15–16, 19
 ACM name, 47
 ACM progress, 33
 computer abstract theory, 29
 computer science definition, 58–69
 computer science funding, 32
 computer science at Stanford, 56–58
 computer science term, 38
 early contributions, 48–50
 Ershov relationship, 215–217
 Feigenbaum trips, 212
 Moler taught by, 215
 road to computer science, 51–56
 scientific background, 50–51
 "Why ACM?", 29n
Forsythe, Warren Ellsworth, 52
Forsythe Hall, 48
Fortran language, 208
Fox, Edward A., 344
FP-6000 mainframe computer, 174
Freed, Roy N., 122n
Funding issues
 ACM accreditation efforts, 97–98
 rejections, 54–55
Furuta, Richard, 343

Gaboury, Jacob, 20
Gaines, Stockton, 312–313
Gallagher, Cornelius, 292, 298
Galler, Bernard
 CBI hosting proposal, 10
 Committee on Computing and the Disadvantaged, 252
 SIC dissolution proposal, 126
 social activism issue, 116, 237
Galvan, Claudia, 160
Gates, Bill, 12
Gavin, James M., 239
Gender. *See* Doctoral women in computer science; Women in computing
General Electric training program, 244
Geographers, women trained as, 73
George and Alexandra Forsythe Papers, 49
German-Swiss-American group, 207
Gerson, Elihu, 134
Gesellschaft für Angewandte Mathematik und Mechanik (GAMM), 210
"Ghetto and the Computer Professional" lecture, 250–251
Giant Brains, or Machines That Think, 2
Gibbs, Norman, 105
Gieryn, Thomas, 28
Gilbarg, David, 53, 56
Gilbert, Jean P., 252, 254–255
Girl Scouts, 157–158
Glazer, Donald, 61
Goker, Ayse, 170
Goldberg, Adele, 78
 ACM president, 85
 places of employment, affiliations, and research areas, 84
 university, dissertation, and first position, 80
Goldwasser, Shafi, 86

Golub, Gene H., 215
Good, Gregory, 30
Google, 158
Gorn, Saul, 37
Gotlieb, Calvin C. "Kelly", 19, 176–179
Gould, Laura, 114
Gouraud, Henri, 281
Government Advisory Committee, 36
Government funding, 25–27, 27n, 33–36, 42–45, 247
Grace Hopper Celebration of Women in Computing, 18
  conferences, 157
  Nadella statement at, 169–170
  producers of, 6
  STEM advocacy, 166–167
Grad, Arthur, 40
Graduate schools findings for doctoral women, 79
Graham, Robert, 300–301
Graham, Susan, 78
  places of employment, affiliations, and research areas, 86
  university, dissertation, and first position, 80
Granville, Evelyn Boyd, 73
Graphic Control and Display of Computer Processes program, 270
"Graphical Man/Machine Communication", 21
Graphics at University of Utah
  community building, 277–284
  introduction, 259–263
  practical applications, 267–272
  problem solving, 272–277
  Salt Lake City in 1966, 263–266
  significant places, 284–285
Gravelle, Paul, 254–255
"Great Society", 41
"Greater Boston Chapter Establishes EDP Training Program for the Underprivileged" article, 250
Green, Venus, 246–247
Griffin, Howard, 254–255
Grosch, Herbert R. J., 132, 235
Gutstein, Lawrence, 249

Hacker debate, 319–323
Haigh, Thomas, 147
Halasz, Frank, 338, 341
Half-tone Graphics Terminal, 273
Hall, Bob, 251
Hall, Wendy, 349
Haltrecht, David, 197
Hammer, Carl, 247
Hamming, Richard
  Commission of Thoughtful Persons, 35n
  Turing laureate, 23
Harlem IBM keypunch operator courses, 251–252
Harrison, Michael A., 128
Hartmanis, Juris, 23–24
Hartree, Douglas, 178
Harvard Mark 3 computer, 4
Harvey Mudd College, 166–167
Haworth, Leland, 43
Hayes, Caroline, 147
Hellman, Martin
  cryptography, 317–318
  Turing laureate, 23
Henkel, Robert, 223–224
Heron, Michael James, 170
Herriot, John G.
  Computation Center, 53
  Computer Science Department, 58
  Ershov relationship, 215–216
  on Forsythe, 48, 56
  Forsythe recruited by, 50–51
Hicks, Marie, 146
Hidden surface problem, 276, 282
Higher Education Act (HEA), 76
Hiltz, Starr Roxanne, 130, 134
History Committee, 6, 12
History of computing
  HOPL effort, 6–10
  study of, 10–13
"History of Programming: Does Our Present Past Have a Future?", 7

History of Programming Languages conferences (HOPL), 6–10
Hoffman, Lance, 310, 327
Holberton, Frances, 2
Hollister, Irene, 113–114
Holt, A. W., 125
Hopper, Grace Murray
  Ada Lovelace Award, 153
  Computer Sciences Man-of-the-Year award, 76
  HOPL conferences, 7
  as mathematician, 73
  "Minerva's Machine: Women and Computing" documentary, 155
  offended by Berkeley, 155n
Horowitz, Ellis, 132
"Hour of Code" program, 158
How Schools Shortchange Girls report, 165
Hudson, Verne, 284
Human rights of computer scientists, 124
Huskey, Harry, 21
  Bendix G-15 computer, 264–265
  computer science curriculum, 94n
  Ershov relationship, 215
"HyperCafe: Narrative and Aesthetic Properties of Hypervideo" paper, 343
HyperCard system, 22, 338
Hypertext. *See* Special Interest Group on Hypertext and the Web (SIGWEB)
Hypertext '87 workshop, 22, 338–339
Hypertext (HT) conference, 340–343, 348
Hypertext on Hypertext product, 346

I Can Be a Computer Engineer, 18, 161
IBM 7044 computer, 273
IBM curriculum support, 94
IEEE, 11
  IEEE Annals of the History of Computing, 262
  Symposium on Security and Privacy, 291
IEEE Computer Society, 26
  description, 5
  discipline definition, 103
  Educational Activities Board, 99–100
  security issues, 288–289
"Imposter problem", 163
Industrial Light and Magic, 283
INFOR: The Canadian Journal of Operational Research and Information Processing, 184, 190
"Information Age: People, Information and Technology" exhibit, 12
Information Processing Techniques Office (IPTO), 260
  Centers of Excellence, 283
  Computer Center, 273
  funding by, 264–266, 271–272
  graphics issues, 270
  peer-review systems, 269
Infrastructure risks, 313–316
Institute for Advanced Study, 31, 176
Institute for Automated Systems in the Building Industry, 119
Institute for Computer Science, 183
Institute for Cybercultural Research, 210
Institute for Numerical Analysis, 50
Institute for the Certification of Computer Professionals (ICCP), 193
Institute for Women and Technology, 152
Institute of Mathematical Statistics, 33
Institute of Mathematics, 213
Institute of Radio Engineers (IRE), 5, 26
Intel, 158
International Algebraic Language, 206
International computing
  cooperation, 210–218
  overview, 203–210
International Conference on Information Processing, 207
International Federation for Information Processing (IFIP), 5, 19
  attendance and discussions, 210
  Canadian representation, 181
  Ershov at, 200, 203–204
  international recognition of computing, 207–208
  as umbrella for many societies, 176

International Federation of Automatic Control Congress, 213
International Geophysical Year (IGY), 205–206
International Organisation for Economic Co-operation and Development (OECD) workshop, 314
Interrupt newsletter
civil rights, 240
CPP membership in (SIC^2), 238
social activism, 126–128
"Issues of War and Racism" article, 240

Jacobs, Jerry, 147
Jervis, Don, 187
Job Opportunities in the Business Sector program, 246
Jobs, Steve, 12
Johns Hopkins University, 242
Johnson, Lyndon B.
computers in education program, 42–43
Great Society program, 41
National Advisory Commission on Civil Disorders, 246
Joint ACM/IEEE conference on digital libraries (JCDL), 348
Joint Task Force on accreditation, 103–104
Journal of Data Management, 188
Journal of the ACM (*JACM*), 19, 178
JOVIAL language, 10
Joyce, Michael, 341

K-12 outreach for women, 157–162
Kahn, Robert, 23
Kansas State University "Hour of Code" program, 158
Kay, Alan, 21
Evans anecdote, 282
HOPL conferences, 7
"Reactive Engine", 268n
recruited by Evans, 281
Turing laureate, 23
Kazarinoff, Nikolai, 224
Keenan, Thomas A., 40n, 41n
Keldysh, M. V., 224–225
Kelemen, Charles, 106
Keller, Evelyn Fox, 75
Keller, Mary Kenneth, 74
Kennedy, John F.
assassination investigation, 131
employment and automation, 233–234
Kennedy, Robert, 116n
Kennedy, Thomas J., Jr., 55
Kerrigan, William, 193
Keypunch operator courses, 251–252
Kierans, Eric, 198
Killiam, Keith, 52
King, John, 122n
King, Martin Luther, Jr., 116n
Kline, Ron, 211
Kling, Robert
CCPP member, 122n
computer crime, 299–300
Computers and Society newsletter, 130
social issues, 132–133
KMS system, 338
Kniefel, David, 95
Knuth, Donald
computer science definition, 14
on Forsythe, 15, 48
Turing laureate, 23
Kocher, Bryan, 323, 325
Kornberg, Arthur, 57
Kouper, Inna, 22
Krementsov, Nikolai, 220

Lacy-Zarubin agreement, 206
Lampson, Butler
information leakage, 304–305
protection views, 301–303
on security specialization, 312
Turing laureate, 23
Landis, Carolyn, 130–131
Laposky, Ben, 284
L'Association Canadienne de l'Informatique, 182
Lavrentiev, M. A., 213, 225
Lazarus, Charles Y., 246

Lederberg, Joshua, 56–57
Leggett, John J., 339, 343
Levin, Benjamin, 121
Levy, Steven, 288
Liapunov, Alexei
  Ershov relationship, 214
  program article, 218
  programming schemata, 205–206
Liberal Arts Computer Science Consortium (LACS), 106
Lichtenberger, Wayne, 266
Licklider, J. C. R., 36, 268–270
Light, Jennifer, 144
Lin, Nora, 159
Linder, Carl, 248
Line Graphics Terminal, 273
Lineback, E. O., 193
Linguistic connections, 337
Link Simulator Company, 262n
Liskov, Barbara
  HOPL conferences, 7
  Turing Award, 23, 86
Longo, Bernadette, 23
Los Alamos National Laboratory hacked, 319
"Los Angeles Chapter Announces Support for Placement of Disadvantaged in EDP Jobs" article, 250
Los Angeles Times on Ershov visit, 222–223
Lovelace, Ada, 153n, 155
Lowell School, 243
Lykos, Peter
  CCPP member, 122n
  challenged by students, 131–132

Macdonald, Neil, 114
Macdonald, Stephen L., 272n
MacDougall, Robert, 196
Machine-independent programs, 206–207
Machover, Carl, 10
MacKenzie, Donald, 288
MacPherson, Don, 192
Madden, Don, 35n
Mahoney, Michael
  history of Cold War technoscience, 202
  HOPL conferences, 7, 10
  marginalization of women, 67–68
"Making metadata: a study of metadata creation for a mixed physical digital collection" paper, 344
Mamrak, Sandra, 78
  places of employment, affiliations, and research areas, 84
  university, dissertation, and first position, 80
Manes, Joan, 240
Mansfield, Mike, 282
Mansfield Amendments, 282
Marchuk, Gurii, 214–216
Marginalization of women, 67–68
Marshall, Catherine C., 344
Mathematical Association of America, 33
Mathematical Sciences program, 31
Mathematical Sciences report, 45
Mathematicians, women trained as, 73
"Matilda effect", 75
Mattel, 159–162
Mauchly, John, 144, 334
Mayer, David B.
  Committee on Computing and the Disadvantaged chair resignation, 255
  Computing Center courses, 252, 255
  Middle West Side Data Processing School, 251
  on motivation, 257
  program funds, 247
  programs for disadvantaged, 254
  SIGCPR chair, 253
  training for disadvantaged, 241–243
McCarthy, John, 15
  Ershov relationship, 215–216, 218
  HOPL conferences, 7
  recruited by Forsythe, 57–58
  Soviet Union visit, 19
  Turing laureate, 23
McClintock, Barbara, 75
McCorduck, Pamela, 53
McCracken, Daniel, 17
  CCPP chair, 122–125

Equal Rights Amendment, 118
SIGCAS supporter, 132
social activism, 113–115
Turchin issue, 121
McDonnell Aircraft Corporation, 54
McIlroy, M. D., 324
McNulty, Kathleen, 144
Meccano differential analyzer, 178
Mechanizing security policies, 304–306
Memex device, 337
MentorNet program, 158
Meritocracy, 169
Mesthene, Emanuel, 234–235
Metropolitan Life Insurance Company, 7
Mezei, Leslie, 182
"Micro-discrimination", 164
Microsoft, 158
Middle West Side Data Processing School, 251
Miller, William, 215–216
"Minerva's Machine: Women and Computing" documentary, 155
Minker, Jack, 121, 124
Minorities in computer science, 164
Minsky, Marvin
Ershov relationship, 215
Turing laureate, 23
Misa, Thomas
computing as discipline, 99
on gender balance, 147–148
history panels, xii, 173n
MITRE Corporation, 252–253
"Model Curriculum for a Liberal Arts Degree in Computer Science", 106
Model curriculum of 1968, 94–95
Moler, Cleve, 215
Moore School of Engineering, ENIAC display, 1
Morgan, M. Granger, 255–256
Morrell, Frank, 52
Morris, Robert T., Jr., 323
Morris worm, 323–326
Morse, Philip M., 37–40, 40n
Morton, Allen L., Jr., 252
Moscow State University (MGU), 205
Moses, Lincoln, 52
Moving Innovation: A History of Computer Animation, 262
Ms. Magazine, 87–88
Mulder, Michael C.
ABET accreditation, 100
on computing curriculum, 91n
discipline definition, 103–104
Mylonas, Elli, 341

Nadella, Satya, 169–170
Name issues, 47–50
Nance, Richard, 121
NASA History Office, 11
National Academy of Engineering (NAE), 157
National Academy of Sciences (NAS)
Committee on Science and Public Policy, 33
Division of Mathematical Sciences, 33
Mathematical Sciences, 45
National Research Council, 34
policy reports on resources, 30
National Advisory Commission on Civil Disorders, 246
National Air and Space Museum, 262n
National Alliance of Businessmen, 246
National Bureau of Standards (NBS)
accessibility control workshop, 305
Institute for Numerical Analysis, 50
security, 288
National Center for the Analysis of Computer Crime Data, 299
National Center for Women and Information Technology (NCWIT), 158
National Committee on Computing and the Disadvantaged, 252–257
National Computer Conference, 168. *See also* AFIPS
National Council of Teachers of Mathematics, 33
National Data Bank, 133
National data center resolution, 296–297
National Institute for Research in Computer Science and Control (INRIA), 281

National Institutes of Health (NIH) funding rejections, 54–56
National Machine Accountants Association (NMAA), 184–185
National Mobilization Committee to End the War in Vietnam, 116n
National Museum of American History, 12, 262n, 289
National Research Council, 24, 34
National Science Foundation (NSF), 13
  Computer Science Education Week, 158
  curriculum funding, 95
  doctoral women in computer science report, 74, 77
  mandate, 27
  Office of Computing Activities, 42–45
  program activities, 30
  status of computing in early 1960s, 31–32
  STEM issues, 157
  Study of Computer Impact on Society and Computer Literacy Courses and Materials grant, 132
National Security Agency (NSA)
  encryption objected to by, 317–318
  SJCC discussions, 292
Naur, Peter
  Algol Bulletin, 207
  HOPL conferences, 7
  Turing laureate, 23
Neidorf, Craig, 326–327
Nelsen, R. Arvid, 20, 297
Nelson, Theodore, 22, 337–338
Networking
  critical infrastructure risks, 313–316
  growth, 311–313
Neugent, Bill, 310–311
Neumann, Peter
  Neidorf case, 327
  reliability and risks study, 316
"New Directions in Operating Systems" workshop, 303
New Security Paradigms Workshop, 328
New York Institute for Technology, 283
Newell, Allen
  on computer science, 14
  computer science letter, 38, 59
  Feigenbaum trained by, 58
  Turing laureate, 23
Nikivincze, Irina, 16
NLS (oN-Line System), 337–338
Norberg, Arthur, 11, 117
November, Joseph, 15–16, 32n
Nuclear weapons operations, 315
Nycum, Susan, 122n

Obama, Barack, 109, 157–158
Oettinger, Anthony G., 13, 26n
  Accreditation Committee appointment, 247
  ACM correspondent hired by, 35
  Committee on Government Relations, 35–36
  on computer school fraud, 249
  on computer science, 32
  funding, 34
  on government funding, 25
  hacker debate, 320
  political function of quantitative studies, 39
  social responsibility, 235–237
Office of Computing Activities, 13, 42–45
Office of Scientific Research and Development, 337
Oldham, Joseph, 133
O'Leary, Dianne, 78
  places of employment, affiliations, and research areas, 84
  university, dissertation, and first position, 80
Oliver, James, 130–131
"On the Social Implications of Computers" article, 126–127
Operating systems security and privacy, 307–308
Operation Bootstrap, 251
Operational Research Group, 184
Organizational boundary-work, 32–36
Organizational Development of the National Science Foundation, 30
Ornstein, Severo, 114

Outskirts of Hope, 254
Outstanding Contribution Award presentation, 112

Panem, Sandra, 75
Papert, Seymour, 215
Parker, Donn B.
  AFIPS study, 314
  computer crime researcher, 298–299
  controls on militant activists, 128
  Executive Committee discussion topics, 239
  hacker debate, 320–321
  Morris worm, 323
  national data center resolution, 296–297
  Neidorf case, 326
Parker, J. D., 192
Parker, Miranda, 161
PDP-8 minicomputer, 273
"Peace and Black Liberation" article, 240
Perlis, Alan J., 13
  ACM archives, 11
  Commission of Thoughtful Persons, 35
  Committee on Government Relations, 36
  on computer science, 14
  computer science funding, 32
  computer science letter, 38, 59
  Ershov relationship, 206–207
  HOPL conferences, 7
  international relationships, 222
  Turing laureate, 2, 23
Perry, Robert, 197
"Personnel Selection & Training of Disadvantaged" presentations, 255
Phong, Bui Tuong, 281
Physics in Canada, 178
Pierce, John R., 41
Pierce Report, 39, 41–43
Pilot ACE project, 21
Pixar, 283
"Plan for Progress", 246
Policies, security, 304–306
Political and social issues debates
  AWC actions, 168–169
  Committee on Computers and Public Policy, 122–125
  conclusion, 135–137
  Equal Rights Amendment, 116–118
  importance, 114–116
  introduction, 111–113
  SIGCAS, 125–135
  Turchin issue, 119–122
"Portrait of South Bronx" film, 249
Preliminary Recommendations for curriculum (ACM 1965), 28
Pribram, Karl, 52
Princeton Institute for Advanced Study, 31
Princeton Symposium on Information Sciences and Systems, 301
Prinz, D. G., 177
Privacy. *See* Security and privacy
Privacy Act, 295
Privacy Protection Study Commission, 293, 295
Problem List of Issues, 135
Professional ethics, 236
Programmers, women as, 144–147
"Programming Costs Too Much" article, 247
Programming languages at HOPL conferences, 6–9
Programming Languages: History and Fundamentals, 7
Programming schemata, 205
Project GENIE time-sharing initiative, 264–266
Project MAC, 266
Project Whirlwind, 206
ProQuest *Dissertations and Theses* database, 77–78
Protection concept, 300–304
"Protection, Security, and Reliability" paper, 303
Public Policy Committee
  McCracken as chair, 115
  on political issues, 112
Public Policy Council, 328
Purdue University
  Division of Mathematical Sciences, 66

Purdue University *(continued)*
early computer department, 73
Purrington, Courtney, 259n

Quarterly Bulletin
CDPSC, 180–182
CIPS, 184, 189
"Quest for realism" in graphics, 277

Race on the Line, 246
Radimsky, Anne-Louise, 78
places of employment, affiliations, and research areas, 84
university, dissertation, and first position, 80
Ralston, Anthony
Committee on Scientific Freedom and Human Rights, 122
human rights of computer scientists, 124
Morgan letters, 255–256
social activism, 114
Ramshaw, Walter, 35n
RAND
Ershov visit, 204, 219–220
Feigenbaum trip, 221–222
Weiner contacts with, 213
"Reactive Engine", 268n
Reagan, Ronald, 315
"Real computing systems", 268
Recoding Gender, 77
Recognition for doctoral women in computer science, 86
Record-keeping security
operating systems, 307–308
privacy-security relationship, 292–295
protection concept, 300–304
security policies, 304–306
SIGCAS establishment, 296–300
Rees, Mina, 73
"Revolution: The First 2000 Years of Computing" exhibit, 12
Richardson, Lewis Fry, 50
Ringland, Kate, 160
Risks for critical infrastructure, 313–316
Risks Forum, 316
Ritchie, Dennis
HOPL conferences, 7
Turing laureate, 23
Robertson, Edward, 132
Role-Based Access Control (RBAC) conference, 328
Rose, Milton E., 42n, 43
Rosser, J. Barkley, 40
Rosser Report, 39–41
Rossiter, Margaret W., 74–75
Rotenberg, Marc, 326
Russell, Andrew, 97
Russell Vernon Anderson Papers, 262n

Sadler, Robert, 255–256
Safeguard antiballistic missile system, 115
SAGE air defense system, 260–261n
Sakharov, Andrei, 119
Salaries for men vs. women, 169–170
Salt Lake City. *See* Graphics at University of Utah
Salton, Gerry, 119–120
Sammet, Jean E., 17
CCPP chair appointment, 123
Equal Rights Amendment, 118
HOPL movement, 6–7
on McCracken, 115
Outstanding Contribution Award presentation, 112
on political issues, 112
privacy issues, 124
Sadler letter, 256
SYMSAM organized by, 76
Turchin issue, 121
San Diego training program, 243–245, 247
Schmittroth, Louis A., 272n
School of Humanities and Sciences, 15
Schure, Alexander, 283
Schweitzer, Jim, 309–310
Schweppe, E. J., 102–103n
Science and gender, 74–75
Science, technology, engineering, and math (STEM) fields
bias against women, 165
female advocacy, 166–167

K-12 outreach for women, 157
women in, 148–149, 151
Sciences of the Artificial, 29
Scientific communities, doctoral women in, 85–86
Scientific Freedom and Human Rights, 124
Scientific identity for computer science
conclusion, 45–46
discipline issues, 28–30
discursive boundary-work, 36–42
establishing, 36–42
introduction, 25–30
Office of Computing Activities, 42–45
organizational boundary-work, 32–36
status of computing at NSF in early 1960s, 31–32
Secureworthiness ratings, 305–306
Security and privacy
conclusion, 328–329
Conference on Computers, Freedom, and Privacy, 326–327
critical infrastructure, 313–316
hacker debate, 319–323
Morris worm, 323–326
operating systems, 307–308
overview, 287–292
privacy concerns, 127, 134
privacy debate, 316–319
privacy-security relationship, 292–295
protection concept, 300–304
security policies, 304–306
SIGCAS establishment, 296–300
SIGSAC, 308–311
specialization and networking growth, 311–313
"Security Controls for Computer Systems", 292
Seitz, Charles, 279
Selwyn, Lee L., 122n
Sexual harassment, 87, 149, 163, 170
Sharp, Ian P., 193
Shaw, Mary, 78
ACM Computing Classification System, 85
places of employment, affiliations, and research areas, 84
university, dissertation, and first position, 80
SIGCOMM group, 308
SIGLINK group, 22, 332, 340–345, 350
SIGMOBILE group, 336
Silicon Valley, 12, 15, 57, 157, 169, 315
naming of, 233
Silver Banquet Committee, 131
Silvey, Ted F., 233
Simon, Herbert
on computer science, 14
computer science letter, 38, 59
Feigenbaum trained by, 58, 212
science characteristics, 29
Turing laureate, 23
Simons, Barbara, 152, 328
Sisakian, N. M., 221
Siscoe, Franc, 221
Sito, Tom, 262
Sketchpad, 275n
Sketchpad: A Man-Machine Graphical Communication System dissertation, 270–271
Slayton, Rebecca, 21, 112
Sloan, Martha, 73
Sloan Foundation, 106
Smith, Alvy Ray, 283
Smith, John B., 338, 341
Smithsonian interviews, 12
Snodgrass, Rick, 12
Social issues. *See* Doctoral women in computer science; Political and social issues debates; Training and education for communities of color; Women in computing
Society for Industrial and Applied Mathematics (SIAM), 5, 33, 86
Society for the History of Technology, 12
Society of Applied Mathematics and Mechanics, 210
Society of Women Engineers (SWE)
on Computer Engineer Barbie, 159
formation, 151

Society of Women Geographers, 151
Solving the Equation, 165–166
SOSP History Day, 23
Southern Regional Education Board, 94
Soviet Academy of Sciences
  Cybernetic Council, 212
  Ershov presentation, 201
  IFIP membership, 207
"Soviet Computer Science, Revisited", 212–213
Soviet Cybernetics: Recent News Items, 219
Soviet Cybernetics Review, 219
Soviet Union
  cold war. *See* Cold war computer science
  Turchin issue, 119–122
Spafford, Eugene
  Morris worm, 323–324
  Neidorf case, 326–327
  treatment of women, 170
Spanish-speaking residents. *See* Training and education for communities of color
Special Committee on the Right of Privacy, 124
Special Interest Committee on Social Concern, 250
Special Interest Committee on Social Implications of Computing [(SIC)^2], 20
  dissolution, 297
  social issues, 125–127, 237–238
Special interest group in programming languages (SIGPLAN), 6–7
Special Interest Group on Business Data Processing (SIGBDP), 21
Special Interest Group on Computer Personnel Research (SIGCPR), 20
  committee for disadvantaged, 252–253
  educational programs, 248
Special Interest Group on Computer Science Education (SIGCSE), 17, 107, 125
Special Interest Group on Computers and Society (SIGCAS), 17, 21
  CCPP coordination, 122, 124–125
  changing interests within, 133–135
  establishment, 296–300
  gender issues, 170
  history, 125–133
  panels submitted to ACM annual conferences, 140–141
  research interests survey, 139
  security and privacy articles, 289–290, 293
  social activism, 113–114
Special Interest Group on Graphics and Interactive Techniques (SIGGRAPH), 10, 21, 260, 270, 283
Special Interest Group on Hypertext and the Web (SIGWEB)
  ACM history, 333–335
  conclusion, 349–353
  history, 335–349
  Hypertext '87 workshop, 338–339
  Hypertext (HT) conference, 340–343
  hypertext visions, 337–338
  introduction, 331–333
  memex proposal, 337
  SIGLINK group, 340–343
Special Interest Group on Information Retrieval (SIGIR), 307
Special Interest Group on Information Technology Education (SIGITE), 125
Special Interest Group on Management of Data (SIGMOD), 307
Special Interest Group on Operating Systems (SIGOPS), 21, 23
  protection concept, 300–304
  security policies, 306
  security and privacy articles, 289–290
Special Interest Group on Programming Languages, 76
Special Interest Group on Security, Audit and Control (SIGSAC), 21, 289, 308–311
Special Interest Group on the Social Implications of Information Processing, 234
Special Interest Group on Training

and Placement of the Culturally Disadvantaged (SIGTAP), 251
Special interest group on urban data systems, planning, architecture, and civil engineering (SIGSPAC), 129–130
Specialization, 311–313
Sprague, Richard, 131
Spring Joint Computer Conference (SJCC), 126, 292
Sputnik launch, 205–206
St. Lawrence Seaway and Power Project, 177
Stallman, Richard, 321
Standards Western Automatic Computer (SWAC), 50
Stanford University, 15
  Akademgorodok Computer Center contacts, 202
  Computer Science Department, 51–52, 56–58
  Forsythe at, 50
  Forsythe Hall, 48
  women at, 16
Star Wars system, 118
Steering Committee, 107
"Stereotype threat", 163
Sterling, Theodore
  CCPP chair, 125
  computer school fraud, 249
  Computers and Society newsletter, 130
Stibitz, George, 334
Stockham, Thomas, 262, 278–279
Stone, Harold, 313
Storyspace system, 338
Stout, Jane, 170
Strachey, Christopher, 177
Strategic Defense Initiative, 118
Stroustrup, Bjarne, 7
Structural factors and doctoral women, 72
Students for a Democratic Society, 116n
Study of Computer Impact on Society and Computer Literacy Courses and Materials, 132
"Study of navigational support provided by two World Wide Web browsing applications" paper, 343
Stuewer, Roger, 11
Subject areas in computer science, 103
Suppes, Patrick, 215
Sutherland, Ivan, 21
  community building, 280–281
  "Computer Graphics: Ten Unsolved Problems" article, 274–276
  Graphic Control and Display of Computer Processes program, 270
  importance of graphics, 271
  move to Salt Lake City, 277–279
  Turing laureate, 23
  Utah history, 262–263, 266–267
Swedish Ministry of Defense study, 314
Syfers, Judy, 87–88
Symposium on Access Control Models and Technologies (SACMAT), 328
Symposium on Document Engineering (DocEng), 348
Symposium on Information Sciences and Systems, 301
Symposium on Large-Scale Digital Calculating Machinery, 333
Symposium on Operating Systems Principles, 304–305
Symposium on Security and Privacy, 21, 291, 311–312
Symposium on Symbolic and Algebraic Manipulation (SYMSAM), 76
Synnoetics, 60
Systers forum, 151–152

Tamarkin, Jacob, 50
Tatarchenko, Ksenia, 19
Tau Beta Pi honor society women's badges, 150–151
Tax status and social activism, 118, 120, 236
Taylor, Robert, 262, 271
Technical Committees in CDPSC, 181
"Technology Gap" symposium, 235
Terman, Fred, 15
  Computer Science Department, 56–59
  computer science skeptic, 53

Thompson, Ken, 23
Title IX clause, 76
Titus, James P., 35, 257–258
Toland, Janet, 17–18, 250, 297
Tomash, Erwin, 10
Toronto chapter of DPMA, 185–186
"Toward an ecology of hypertext annotation" paper, 344
Training and Curriculum Committee, 244
Training and education for communities of color
  ACM discussions, 232–240
  ACM involvement, 248–252
  computer school fraud, 249
  conclusion, 257–258
  introduction, 229–232
  National Committee on Computing and the Disadvantaged, 252–257
  programs, 240–248
Transactions on Database Systems (*TODS*), 307
Trans-Canada Telephone System, 196
Transnational mediation. *See* Cold war computer science
Trustworthy Computer System Evaluation Criteria (TCSEC), 305–306
Tucker, Allen, 105
Tupac, Jim, 247
Turchin, Valentin, 115, 119
Turchin issue, 115, 119–122
Turing, Alan, 2, 21
Turing Award, 12
  initial, 2
  recipients, 23
  Sutherland, 266
  women, 86
Turn, Rein
  AFIPS study, 314
  on authentication, 308
  privacy symposium, 311
Turner, Len
  on CDPMAI, 193, 196–197
  Journal of Data Management, 188
Turoff, Murray, 130
Ulery, Dana, 78
  places of employment, affiliations, and research areas, 84
  university, dissertation, and first position, 80
"Ultimate compromise" for discipline definition, 102
Unemployment and automation, 233–235
UNESCO, 207
UNIVAC 1108 computer, 273
Univac company, 1
University computer science degrees, 26
University of Alberta, 183
University of California at Irvine women career advancement, 152
University of California–Berkeley
  computer-science degrees for women, 152
  computer science major, 105
University of Missouri–Rolla, 131–132
University of Pennsylvania, ENIAC display, 1
University of Toronto ACM meeting, 176–178
University of Utah graphics. *See* Graphics at University of Utah
University of Waterloo, 183
University of Western Ontario, 183
University of Wisconsin–Madison, 74
"University's Educational Program in Computer Science" manifesto, 60–61
Urban Education Committee
  computer operator program, 251
  impediments to education, 244
Urban League Training Center, 250
Urban Symposium, 249
"URGENT—An Increased Dialog with Society" article, 239
USACM (ACM US Public Policy Council), 125
Use of Computers to the Benefit of Mankind committee, 125
"Use of English as a Programming Language" article, 6

USENIX, 5
Utechnic, 274

Van Atta, Bruce, 122n
van Dam, Andy, 10, 22, 339
Vanishing paper trails, 309
Vardalas, John, 197
Very Large Data Bases (VLDB), 307
Vietnam war
  ACM controversy over, 115–116
  Computers and Society newsletter, 131
  CPP opposition to, 127
  financial pressure from, 43n
  petition to oppose, 237
  polarization from, 111, 297
"Vital Factor in Race: US Leading Russia in Space Computers", 222–223
von Neumann, John, 209
  ACM proposal opposed by, 334
  Princeton computer, 31

Waelsch, Salome, 75
Walker, Gianna, 269–270
Walker, Henry, 106
Walker, Janet, 339
Walker, Virginia, 168–169
Warburton, Peter, 234
Ware, Willis
  AFIPS study, 314
  Congressional testimony, 316–317
  hacker debate, 320
  privacy issues, 124, 292–295
  Soviet Union visit, 219
Warfield, Henry, 240
Wargames film, 319
Warner, Homer K., 272n
Warnock, John, 21
  dissertation, 281–282
  Evans as motivator, 279–280
Warnock Engineering Building, 259n
Warren, Jim, 327
Waterman, Alan T., 39
Watson, Richard, 215
Web science, 349
Weiner, Norbert, 213
Weingarten, Frederick W., 27n, 44–45n
Weissman, Clark, 305
Weizenbaum, Joseph, 114, 122n
Westin, Alan, 130, 292, 296
White House science fairs, 157–158
Whitehead, E. James, 352
Whitney, James, 260n
Whitney, John, 260n
Whorf, Benjamin Lee, 337
"Why ACM?" letter, 29n
"Why I want a wife" article, 88n
Why So Few, 165
Wilkes, Maurice
  Ferranti Mark I installation, 177
  Turing laureate, 23
Will, Daniel A., 186
Williams, B. O.
  on Rosser Report, 41n
  on von Neumann grant, 31n
Williams, Greg, 114
Williams, Samuel, 334
Winograd, Terry, 327
Wirth, Niklaus
  Ershov relationship, 215
  HOPL conferences, 7
  Huskey relationship, 215–216
  Turing laureate, 23
Wise, David, 12
Women in Cell Biology, 151
Women in computing
  conclusion, 171–172
  doctoral women. *See* Doctoral women in computer science
  introduction, 143–150
  K-12 outreach, 157–162
  marginalization of, 67–68
  organized advocacy, 150–157
  shifting nature of advocacy, 162–171
  Stanford University, 16
  women's movement, 18
Women's Day, 167
Women's Service Committee, 151
Woodger, Mike, 209

"Working in Engineering and Computer Science: A Conference for Women", 152
World War II, women computing roles in, 144
World Wide Web, 22, 342–343, 345
Worsley, Beatrice, 177
Wright, Rosemary, 147
Xanadu system, 338
"Xerox's NoteCards, Owl's Guide, and especially Apple's HyperCard", 22

Yost, Jeffrey, 262
Young, Michael, 259n
Young, Whitney M., Jr., 248n
Youth International Party (Yippies), 116n

Contributor Biographies

Thomas Misa directs the Charles Babbage Institute (http://www.cbi.umn.edu/), a leading international research and archiving center specializing in the history of information technology, at the University of Minnesota. There he teaches in the Program for the History of Science, Technology, and Medicine and holds the ERA Land Grant Chair in History of Technology in the Department of Electrical and Computer Engineering. From 2014 to 2016 he was chair of the ACM History Committee, which assisted in the support and preparation of these chapters.

Janet Abbate is an associate professor of Science and Technology in Society at Virginia Tech and serves as Co-director of the STS graduate program in Northern Virginia. Her research has focused on the history, culture, and policy of computers and the Internet. Her books include *Inventing the Internet* (1999) and *Recoding Gender* (2012). Her current research, which was supported by a 2012 ACM History Fellowship, investigates the emergence of computer science as an intellectual discipline, an academic institution, and a professional identity.

Amy Sue Bix is Professor of History at Iowa State University and director of ISU's Center for Historical Studies of Technology and Science. Her book in progress is *Recruiting Engineer Jane and Astrophysicist Amy: American STEM Advocacy for Girls, 1965–2015*. Bix's 2013 book *'Girls Coming to Tech!': A History of American Engineering Education for Women* (MIT Press) was awarded the 2015 Margaret W. Rossiter History of Women in Science Prize from the History of Science Society.

Scott Campbell directs the Centre for Society, Technology and Values (http://cstv.uwaterloo.ca), and is a Lecturer in the Department of Systems Design Engineering at the University of Waterloo. He teaches courses about the relationship between technology and society and the history of computing. His main research interest is the history of computing and computer science in Canada and he is currently working on a book about the University of Waterloo's Faculty of Math as part of the faculty's 50th anniversary celebration in 2017.

Sebastian Dziallas is a Ph.D. student in the Computing Education Research Group in the School of Computing at the University of Kent. His research examines characteristics of computer science graduates and uses a highly qualitative, narrative methodology to explore the sense graduates make of their own educational experiences within their wider learning trajectories. Before joining the school, he graduated from Olin College of Engineering in the United States.

Sally Fincher is a professor of Computing Education in the School of Computing at the University of Kent, where she leads the Computing Education Research Group. Her work is centrally concerned with the teaching and learning of computing, with particular emphasis on teachers and teaching practices. She has worked on several major computing education projects, such as the *Bootstrapping Research in Computer Science Education* series, and the UK "sharing practice" project: http://www.sharingpractice.ac.uk. She is Editor-in-Chief of the journal *Computer Science Education* (jointly with Laurie Murphy), a UK National Teaching Fellow, a Senior Fellow of the UK Higher Education Academy, and a Fellow of the Royal Society of Arts.

Jacob Gaboury is an assistant professor of Digital Media & Visual Culture in the Department of Cultural Analysis and Theory at Stony Brook University, SUNY. His work engages the history and critical theory of digital media through visual culture and media theory, and his current book project offers an archaeology of early computer graphics at the University of Utah in the 1960s and 1970s. In the past he has held fellowships through the Charles Babbage Institute, the ACM, the IEEE, the Smithsonian Institution, and the Max Planck Institute for the History of Science.

Inna Kouper is a research scientist and assistant director of the Data to Insight Center at Indiana University Bloomington (d2i.indiana.edu). Her research interests are in the history and sociology of knowledge production and dissemination, with a particular emphasis on the emerging technologies and data practices. In 2011 she received the ACM History Committee fellowship, which supported research for Chapter 13.

R. Arvid Nelsen is the Rare Books & Manuscripts Librarian for the Bridwell Library at Southern Methodist University. As the Curator & Archivist for the Charles Babbage Institute, 2007–2010 and 2012–2016, he created the Social Issues in Computing

Collection which documented computing and its impact from the perspectives of race, ethnicity, gender, and sexuality with emphases on employment, education, the environment, warfare, privacy, and security. He also collected such notable organizations and individuals as the Association for Computing Machinery, Stephen J. Luksaik, the Gartner Group, James W. Cortada, Alias|Wavefront, Lockheed Martin, Carl Machover, Erwin and Adelle Tomash, and Michael S. Mahoney.

Irina Nikivincze (aka Nikiforova) is a postdoctoral researcher at the Center for Accessible Materials Innovations (CAMI), an initiative of the AMAC Accessibility Solutions and Research Center at the Georgia Institute of Technology where she studies the impact of computers and assistive technologies on academic achievements of students with disabilities. Irina's doctoral dissertation focused on the achievements and recognition of A.M. Turing Award winners, the recipients of the ACM's "Nobel Prize" in computing. Since most of them were men, Irina explored the experiences and achievements of doctoral women in computer science in her post-graduate work. In 2013 Irina received a fellowship from the ACM History Committee to interview some of these women. The findings from these interviews are reported in this book.

Joseph November is associate professor and McCausland Fellow in the Department of History at the University of South Carolina. He has been examining the history of computer use in biology and medicine. His first book, *Biomedical Computing: Digitizing Life in the United States* (Johns Hopkins University Press) was awarded the Computer History Museum Prize in 2013. Funds provided by the ACM supported the research conducted for his chapter in this book.

Rebecca Slayton is an assistant professor in the Department of Science & Technology Studies and the Judith Reppy Institute for Peace and Conflict Studies, both at Cornell University. Her research examines the relationships between and among technological expertise, governance, and risk, with a focus on international security and cooperation since World War II. She is author of *Arguments that Count: Physics, Computing, and Missile Defense, 1949–2012* (MIT Press, 2013), which shows how the rise of software engineering helped make the risks of complex software more visible in public discourse and policy about missile defenses. She is currently working on a book about the history of cybersecurity expertise. Her contribution to this book was partially supported by a grant from the ACM History Committee.

Ksenia Tatarchenko is a lecturer at the Global Studies Institute, Geneva University, specializing in the history of Russian science and technology. She has held positions as a visiting assistant professor of History at NYU Shanghai and a postdoctoral fellow at the Harriman Institute, Columbia University. Tatarchenko held the ACM History Fellowship in 2011. She studies the Soviet Information Society, transnational knowledge circulation, and IT transfer. She has published on the history of the International Federation for Information Processing and the Soviet-French cooperation in computing. She is a co-editor of the forthcoming special issue of *Historical Studies in Natural Sciences* devoted to Cold War mathematics.

Janet Toland is a senior lecturer in Information Systems at Victoria University of Wellington, in New Zealand. Her research interests include history of computing, green information systems, and community informatics. For her Ph.D. research she used historical methods to investigate ICT use in two regions of New Zealand. In 2015 she received an ACM History Fellowship and spent five months at the Charles Babbage Institute, University of Minnesota collecting data on ACM's role in influencing public policy around issues related to computing and society.