History of Computing

Bernadette Longo

Words and Power

Computers, Language, and U.S. Cold War Values

Springer

History of Computing

The *History of Computing* series publishes high-quality books which address the history of computing, with an emphasis on the 'externalist' view of this history, more accessible to a wider audience. The series examines content and history from four main quadrants: the history of relevant technologies, the history of the core science, the history of relevant business and economic developments, and the history of computing as it pertains to social history and societal developments.

Titles can span a variety of product types, including but not exclusively, themed volumes, biographies, 'profile' books (with brief biographies of a number of key people), expansions of workshop proceedings, general readers, scholarly expositions, titles used as ancillary textbooks, revivals and new editions of previous worthy titles.

These books will appeal, varyingly, to academics and students in computer science, history, mathematics, business and technology studies. Some titles will also directly appeal to professionals and practitioners of different backgrounds.

More information about this series at http://www.springer.com/series/8442

Bernadette Longo

Words and Power

Computers, Language, and U.S. Cold War Values

Springer

Bernadette Longo
New Jersey Institute of Technology
Newark, NJ, USA

ISSN 2190-6831 ISSN 2190-684X (electronic)
History of Computing
ISBN 978-3-030-70372-1 ISBN 978-3-030-70373-8 (eBook)
https://doi.org/10.1007/978-3-030-70373-8

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Acknowledgment

When the idea for this book was first conceived in 2019, the word *pandemic* brought to mind images of Spanish flu patients in medical wards after World War I. Now I have been living in isolation for over 9 months because of the COVID-19 pandemic. Many of the plans I had for this book needed to be reconsidered because of travel bans, business closures, and health risks. Life has changed, but one constant has been the story that I wanted to tell in this book.

A number of people have been instrumental in bringing this story to completion: Peggy Kidwell at the National Museum of American History has supported my humanistic approach to technology studies for nearly two decades and continues to give me encouragement. I want to mention her 1998 article “Stalking the Elusive Computer Bug” in the *IEEE Annals of the History of Computing* as a kindred study of the importance of establishing a shared terminology for early computer developers, even though I did not cite it directly in this book.

Archivist Nathan Brewer at the IEEE History Center digitized materials in their collection for me while their workplace was closed due to the pandemic. Those materials were enlightening on the personal relationships that shaped the IRE glossary work in the late 1940s and early 1950s. In a period when I was unable to visit those collections in person, Nathan enabled my work to continue.

While the archive closures continued, David Brock and Dag Spicer at the Computer History Museum, and Erik Rau and Linda Gross at the Hagley Museum also generously digitized materials and sent them to me to inform the story in this book. Max Campbell at IBM Corporate Archives was also helpful in searching materials for me while I was not able to visit the archives in person. It has been heartening to collaborate with these colleagues in a virtual world while our real world has been mostly shut down.

The idea for this book originated in an earlier world when I could travel to archives to work with boxes of materials in person. It grew out of a biography of computer pioneer and social activist Edmund Berkeley that I was working on at the Charles Babbage Institute. Throughout the years that I worked on this project and even when it was doubtful that I would ever bring it to completion, Tom Misa and

Jeff Yost never gave up on that project. The reception of that book encouraged me that there is a place for humanistic and cultural studies in the world of computer history.

Newark, NJ, USA Bernadette Longo

Contents

Abbreviations

ABC	Atanasoff-Berry Computer
ACE	Automatic Computing Engine
ACM	Association for Computing Machinery
AEC	Atomic Energy Commission
AEC GAC	Atomic Energy Commission General Advisory Committee
AIEE	American Institute of Electrical Engineers
AMD	US National Bureau of Standards Applied Mathematics Division
AMP	US National Bureau of Standards Applied Mathematics Panel
BCD	Binary Coded Decimal
BINAC	Binary Automatic Computer
BRL	Ballistics Research Laboratory
C3S	ACM Curriculum Committee on Computer Science
CBS	Columbia Broadcasting System
CPU	Central Processing Unit
CRC	Computer Research Corporation
EDSAC	Electronic Delay Storage Automatic Computer
EDVAC	Electronic Discrete Variable Automatic Computer
ENIAC	Electronic Numerical Integrator and Computer
ERMA	Electronic Recording Machine, Accounting
IAS	Institute for Advanced Study in Princeton, New Jersey
IBM	International Business Machine Corporation
IFF	Identification, Friend or Foe
INA	US National Bureau of Standards Institute for Numerical Analysis
IRE	Institute of Radio Engineers
IWE	Institute of Wireless Engineers
MANIAC	Mathematical Analyzer Numerical Integrator and Computer
MIDAC	Michigan Digital Automatic Computer
MIT	Massachusetts Institute of Technology

MTAC Committee	National Research Council Committee on the Bibliography of Mathematical Tables and Other Aids to Computation
NBS	US National Bureau of Standards
NCR	National Cash Register Corporation
NIST	National Institute of Standards and Technology
NOTS	Naval Ordnance Testing Station
NPL	National Physical Laboratory, England
NRDC	National Defense Research Council
NSF	National Science Foundation
OCR	Optical Character Recognition
OSRD	Office of Scientific Research and Development
RAYDAC	Raytheon Digital Automatic Computer
RCA	Radio Corporation of America
SAGE	Semi-automatic Ground Environment
SEAC	Standards Eastern Automatic Computer
SSEC	IBM Selective Sequence Electronic Calculator
SWAC	Standards Western Automatic Computer
UCLA	University of California, Los Angeles
UN	United Nations
UNAEC	United Nations Atomic Energy Commission
UNIVAC	Universal Automatic Computer
UNO	United Nations Organization
US	United States
USN	United States Navy
USO	United Service Organizations
USSR	Union of Soviet Socialist Republics

Chapter 1
Introduction

Abstract This introduction provides a rationale for a humanistic study of computer history based on philosophical theories of Michel Foucault and François Lyotard. It also explores questions about how one possible definition was produced and legitimated while other possible definitions were not legitimated, even though they may have been produced. The introduction finally discusses the educational role of a type of dictionary – glossaries – in establishing a professional community around the development and operation of electronic computers.

> "*The manner in which sense perception is organized, the medium in which it is accomplished, is determined not only by nature but by historical circumstances as well.*" *Walter Benjamin, "The work of art in the age of mechanical reproduction," 1936* [3]

> "*Good technical writing is so clear that it is invisible. Yet technical writing is the mechanism that controls systems of management and discipline, thereby organizing the operations of modern institutions and the people within them.*" *Bernadette Longo, "Spurious Coin," 2000* [11]

> "*Scientific knowledge, like language, is intrinsically the common property of a group or else nothing at all. To understand it we shall need to know the special characteristics of the groups that create and use it.*" *Thomas Kuhn, "The Structure of Scientific Revolutions," 1970* [9]

1.1 Why Words Matter

For those of us who carry smart-phone computers in our pockets, it is difficult to imagine a time when there were only a handful of computers in the world. Yet I know a colleague who, as a student, liked to study at a desk inside one of these computers at his university because it was quiet and solitary in there. Yes, he studied at a full-sized desk **inside** the computer.

When computers were room-sized machines, there was no discipline called "computer science." There were mathematicians and electrical engineers and physicists who dreamed of "mechanical brains." And then they built large-scale automatic calculating machinery. That's what these machines were called in the mid-twentieth century. In those early days of computer development, people worked

B. Longo, *Words and Power*, History of Computing,
https://doi.org/10.1007/978-3-030-70373-8_1

in laboratories that were isolated from one another. There was no standardized terminology that those people used to communicate information about their computer development projects. There were no established communication channels to facilitate information sharing. This is a story of how computer people developed that body of specialized terminology and established those communication channels as important building blocks for creating the distinct discipline and profession we now call computer science.

This is also a story of how these computer people worked within an international context of hot and cold wars – how those international relationships shaped perceptions of their work and the products of that work. At the center of this story are the people who first imagined large-scale automatic calculating machines and worked together to create them. Early on, they were motivated by the rapidly growing telephone industry and its demand for complex number calculations that were needed for expansion of long-distance telephony. Technology development was outstripping human calculating ability.

Computer developers were then motivated by the military imperative to generate ballistic firing tables more quickly than was possible by human computers working with pencils, papers, and desktop mechanical calculators. As World War II ground on, the expansion of military theaters into more geographic areas with specific atmospheric conditions was threatening the Allies' ability to dominate Axis forces. Ultimately, the computers that were developed for calculating ballistic firing tables were put to use to help physicists analyze questions about thermonuclear explosions and the feasibility of dropping the first atomic bomb, which subsequently led to the end of World War II and the beginning of the Cold War. As World War II ended and a new kind of ideological/psychological war began, the computer developers, who had been working in isolation behind laboratory walls secured by information restrictions, came together with a new urgency to develop "mechanical brains" that would help to protect the Free World against the threat of Communism. The world also faced the threat of international thermonuclear warfare, along with the possibility of harnessing atomic power for generating electricity and prosperity. The urgent threat and the unfulfilled promise of the Atomic Age would require mathematical calculating ability that exceeded that of human computers. This new age required new machines that mimicked human calculations but worked much faster than humans. Computer people shouldered their responsibilities for developing these machines and shaping human relations in the Atomic Age.

1.2 What People Tell Us About Computers

Although computer histories are often told in heroic terms, smaller stories of human relations underpin these tales of hardware and software development. The decisions that computer people made within their institutional and social contexts shaped the paths of technology development as they built room-sized digital computers with thousands of vacuum tubes. The actions of computer developers after World War II influenced the trajectory of technology development and professionalization through the Atomic Age and beyond. The actions of these early computer people continue shape the human-computer interactions that we expect from our intelligent machines today.

When computer histories foreground innovations in hardware and software – in what Michael Mahoney (1988) called "'insider' history full of facts and firsts" [14, p. 114] – these stories minimize the social contexts in which people made decisions and took actions that contributed to these innovations. Without these contexts, the progress of technological innovation can seem inevitable rather than localized and tentative. These "insider" accounts do provide firsthand knowledge of computer development from one perspective but are limited "by the current state of knowledge and bound by the professional culture" [14, p. 114]. Authors who lived these histories firsthand might take their particular and localized states of knowledge as "givens … [but] a more critical outside viewer might see [these] as choices" [14, p. 114] among equally possible alternative paths. From an "insider" perspective, choices made by people relating to the development of electronic computers can be seen as inevitable steps in technological progress. From this worldview, they acted within an objective realm of pure and applied science – a realm free of politics and culture.

This path of technology development from large-scale automatic calculating machines to smart phones was not inevitable but instead reflects the politics and cultures of specific locations and times along the way [11, 13]. As Tom Misa (2007) argued, the actions of computer people bring about cultural – as well as technological – changes. This is why Misa advocated that histories of computer development should include the social and institutional influences impacting people who worked on these machines and their programming [17, p. 54–56]. He also foresaw that studying the history of computing in contexts of "broad historical transformations" would necessarily require historians to draw on a "wider set of research methods" than used to write more decontextualized histories of technology development [17, p. 59]. Following Misa's advice, human-centered stories of computer development and biographies of computer people can contribute to developing histories of computing machinery that encompass broad historical transformations, both cultural and technological.

In his overview of the state of computer historiography, James Cortada (2015) found that human-centered stories of computer development "have been slow to appear" despite the maturation of the field [4, p. 27]. He noted, though, that this humanities-based approach to computer history resulted in studies that "emphasize the role of specific individuals in shaping development and use of computing" [4, p. 27]. The study that follows here responds to Cortada's call to investigate how the actions of specific individuals shaped the development and use of computers, as well as the development of computer science as a profession. In particular, this study looks at how the efforts of early computer people to establish a standardized nomenclature for their field helped them to respond to the need for rapid technology development in the face of Cold War national security concerns. This nomenclature allowed for information sharing among people from different laboratories who had worked in isolation during World War II. It also provided a foundation for developing computer literature that was necessary for the growth of computer science as a profession separate from mathematics, electrical engineering, and physics.

1.3 What Technical Language Tells Us About People

As I have argued elsewhere [11, 12], technical language is the mechanism that people employ to turn knowledge into cultural capital or social value. Rather than being a neutral conduit to transport information from one point to another in a positivist sense, technical language mediates the transfer from an applied scientist or computer developer to other developers or end users. Through this mediation, technical language serves an active role in knowledge creation within social contexts. In the case of early computer development, people designing these mechanical and electronic calculating machines initially lacked a common body of specialized terms to describe and communicate information about their work to other people. They relied on analogy and terminology from other fields, such as electrical engineering or psychology, to represent ideas about computing machines. At first, terms were specific to individual laboratories and the people working in them. As computer developers communicated with each other more widely after World War II security clearances relaxed, idiosyncratic terms were standardized through collaboration and contest within institutions.

Language is how we give voice to technical knowledge that participates in systems of institutional power. It is shaped by these societal systems, while simultaneously shaping them. In this current study, computer terminology was initially a contested site of knowledge production as people came together from their isolated workplaces with a common goal of rapid computer development. Whose knowledge would prevail? Who would claim the power to define terms that would become authoritative in a new industry and profession that was shaping social, political, and economic relations on an international scale? Debate about these knowledge production questions took place within military, academic, and industrial institutions. Some knowledge would be legitimated through standardized terminology, such as knowledge about electronic computing machines and programming. Other possible knowledge would be marginalized, such as knowledge about analog and other mechanical calculating machines. In these debates, institutions themselves can be seen as cultural agents influencing discourse and professional development. Vincent Leitch (1992) described how institutions act as cultural agents to legitimate and reward knowledge made through standardized technical language:

> Through various discursive and technical means, institutions constitute and disseminate systems of rules, conventions, and practices that condition the creation, circulation, and use of resources, information, knowledge and belief. Institutions include, therefore, both material forms and mechanisms of production, distribution and consumption and ideological norms and protocols shaping the reception, comprehension, and application of discourse. … Institutions often enable things to function, inaugurate new modes of knowledge, initiate productive associations, offer assistance and support, provide useful information, create

> helpful social ties, simplify large-scale problems, protect the vulnerable, and enrich the community. [10, p. 127–128]

Because institutions are cultural agents that affect discourse practices, recognition of organizations' participation in cultural contexts enables a study that can illuminate assumptions about the inevitable roles of technical language in a specific culture at given historical moments – roles such as information mediator or professional foundation builder.

This study traces the development and standardization of computer terminology in the United States from the 1940s into the 1960s. Its method of inquiry has heeded Thomas Kuhn's (1970) call for historians of science to "display the historical integrity of that science in its own time" [9, p. 3]. In this vein, I have attempted to reconstruct a cultural context for past language practices within a field that would become computer science in order to understand these past practices not as ill-fitting or quaint compared to contemporary understanding, but as legitimate practices within their situated historical contexts. Since technical language deals in knowledge made through pure and applied science, the practice of communicating this knowledge can be seen as a scientific mechanism or apparatus for determining proper valuation and credit for the product, in this case computers. By communicating their knowledge, scientists and technology developers sought to modify the scriptures of their field and, thereby, the concepts that regulate further knowledge production. If a person's or a committee's communication could modify these concepts in ways that could be translated into technological advances, that knowledge was accorded value. This ability to transform knowledge into value is central to the function of technical language.

Translating language into technological advances is not merely a collaborative effort but also involves contests for cultural capital. "Making sense" to the winners of these contests may not agree with the "common sense" of others, whose language and knowledge was delegitimated. Jean-François Lyotard (1988) described this silencing of devalued knowledge as a "wrong" suffered in "a case of conflict between (at least) two parties, that cannot be equitably resolved for lack of a rule of judgment applicable to both arguments" [13, p. xi]. In the case of early computer development, there were no mutually agreed-upon rules for equitable judgment in cases of disputed definitions for what would become the *lingua franca* of a new profession called computer science. In the absence of rules of equitable judgment, decisions about whose discourse would prevail must privilege one group's knowledge production over other possible ways of making knowledge. Unlike a simple idea of collaboration, Lyotard's theory of knowledge production through discourse legitimation holds that power is unevenly distributed among possible ways of knowing. Discourse becomes a site of contests for knowledge legitimation and cultural advantage. Technical language participates in these struggles by assigning value to legitimated knowledge as the currency of a scientific knowledge economy. Devalued knowledge and its associated technical language will not circulate in this economy at full cultural value.